Technische Hochschule Nürnberg Georg Simon Ohm
Fakultät efi

Studiengang Bachelor Media Engineering
Vertiefungsrichtung Medientechnik

Bachelorarbeit von

Tim Abraham
Matrikel-Nr. 3022687

Konzeption und prototypische Implementierung einer Infrastruktur für Wissenstransfer in einer Tribe-basierten Organisation unter Einbindung von Micropayments als Anreizsystem

Sommersemester 2020
Abgabedatum: 23.05.2020

Betreuer:
Prof. Dr. O. Hofmann
Dipl.-Inf. S. Blümm
adorsys GmbH & Co. KG

Schlagworte: Blockchain, Micropayments, DRM, Wissenstransfer, Tribe-Organisation

Hinweis: Diese Erklärung ist in alle Exemplare der Abschlussarbeit fest einzubinden. (Keine Spiralbindung)

TECHNISCHE HOCHSCHULE NÜRNBERG
GEORG SIMON OHM

Prüfungsrechtliche Erklärung der/des Studierenden

Angaben des bzw. der Studierenden:

Name: Abraham Vorname: Tim Matrikel-Nr.: 3022687

Fakultät: Elektro-, Feinwerk-, Informationstechnik Studiengang: Media Engineering

Semester: Sommersemester 2020

Titel der Abschlussarbeit:

Konzeption und prototypische Implementierung einer Infrastruktur für Wissenstransfer in einer Tribe-basierten Organisation unter Einbindung von Micropayments als Anreizsystem.

Ich versichere, dass ich die Arbeit selbständig verfasst, nicht anderweitig für Prüfungszwecke vorgelegt, alle benutzten Quellen und Hilfsmittel angegeben sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

Nürnberg, 22.05.2020, Tim Abraham

Ort, Datum, Unterschrift Studierende/Studierender

Erklärung zur Veröffentlichung der vorstehend bezeichneten Abschlussarbeit

Die Entscheidung über die vollständige oder auszugsweise Veröffentlichung der Abschlussarbeit liegt grundsätzlich erst einmal allein in der Zuständigkeit der/des studentischen Verfasserin/Verfassers. Nach dem Urheberrechtsgesetz (UrhG) erwirbt die Verfasserin/der Verfasser einer Abschlussarbeit mit Anfertigung ihrer/seiner Arbeit das alleinige Urheberrecht und grundsätzlich auch die hieraus resultierenden Nutzungsrechte wie z.B. Erstveröffentlichung (§ 12 UrhG), Verbreitung (§ 17 UrhG), Vervielfältigung (§ 16 UrhG), Online-Nutzung usw., also alle Rechte, die die nicht-kommerzielle oder kommerzielle Verwertung betreffen.

Die Hochschule und deren Beschäftigte werden Abschlussarbeiten oder Teile davon nicht ohne Zustimmung der/des studentischen Verfasserin/Verfassers veröffentlichen, insbesondere nicht öffentlich zugänglich in die Bibliothek der Hochschule einstellen.

Hiermit ☒ genehmige ich, wenn und soweit keine entgegenstehenden Vereinbarungen mit Dritten getroffen worden sind,

☐ genehmige ich nicht,

dass die oben genannte Abschlussarbeit durch die Technische Hochschule Nürnberg Georg Simon Ohm, ggf. nach Ablauf einer mittels eines auf der Abschlussarbeit aufgebrachten Sperrvermerks kenntlich gemachten Sperrfrist

von 0 Jahren (0 - 5 Jahren ab Datum der Abgabe der Arbeit),

der Öffentlichkeit zugänglich gemacht wird. Im Falle der Genehmigung erfolgt diese unwiderruflich; hierzu wird der Abschlussarbeit ein Exemplar im digitalisierten PDF-Format auf einem Datenträger beigefügt. Bestimmungen der jeweils geltenden Studien- und Prüfungsordnung über Art und Umfang der im Rahmen der Arbeit abzugebenden Exemplare und Materialien werden hierdurch nicht berührt.

Nürnberg, 22.05.2020, Tim Abraham

Ort, Datum, Unterschrift Studierende/Studierender

Formular drucken

In der folgenden Arbeit wird aus Gründen der besseren Lesbarkeit überwiegend das generische Maskulinum verwendet. Alle personenbezogenen Formulierungen beziehen sich auf Personen beiderlei Geschlecht.

Abstract

The aim of this thesis is to design a proof-of-concept of an infrastructure for a company internal domain-specific knowledge transfer system based on a blockchain. On the basis of blockchain bound proprietary tokens and smart contracts as DRM control, a content system can be balanced by micropayments according to on-demand principles. The incentive system created by the infrastructure should help companies to improve the knowledge transfer for employees and to save resources. The infrastructure consisting of the coupling of a normal DRM system and a blockchain combines the advantages of both systems together. The secure and easy payment for the provided data is crucial for the end user. Further development steps are required before such a system can be adapted for distinct usage environments. Nevertheless, the presented concept can work and contributes a part to the development of the often limited usability of blockchain systems.

Kurzfassung

Ziel der Arbeit ist es ein „Proof-of-Concept" einer Infrastruktur für ein firmeninternes domänenspezifisches Wissenstransfersystem auf Basis einer Blockchain zu entwerfen. Auf Basis von blockchaingebundenen proprietären Tokens und Smart Contracts als DRM-Steuerung, kann ein Content-System angelehnt an On-Demand-Prinzipien durch Micropayments bilanzierbar werden. Das durch die Infrastruktur geschaffene Anreizsystem soll Firmen helfen für Angestellte den Wissenstransfer besser und ressourcenschonender zu gestalten. Die Infrastruktur bestehend aus der Kopplung eines normalen DRM-Systems und einer Blockchain verbindet die Vorteile der beiden Systeme miteinander. Für den Endanwender ist die sichere und einfache Bezahlung der bereitgestellten Daten ausschlaggebend. Bis zur Adaption eines solchen Systems für bestimmte Nutzungsumgebungen sind noch weitere Entwicklungsschritte zu bewältigen. Dennoch funktioniert das vorgestellt Konzept und trägt einen Teil zur Entwicklung der oft eingeschränkten Benutzerfreundlichkeit von Blockchainsystemen bei.

Inhaltsverzeichnis

1 Einleitung

Das starke Wachstum in der IT-Branche und der Trend zu agilen Arbeitsweisen stellen Unternehmen vor organisatorische Herausforderungen. Zum einen vergrößert sich der Mitarbeiterpool, oft um Personen mit wenig projektspezifischer Berufserfahrung, zum anderen findet ein Entwicklungsprozess von klassischen Organisationsstrukturen hin zu kleineren agilen Einheiten statt. Doch der Wissenstransfer und die Kommunikation unter Angestellten wird über agile Strukturen hinweg aufwändiger. Daraus resultiert ein unvorteilhaftes Verhältnis zwischen Junioren und erfahrenen Mitarbeitern, welches beispielsweise das klassische „On-The-Job-Training" verkompliziert und zeitaufwändiger macht. Der mittelständische IT-Dienstleister „adorsys GmbH & Co. KG" durchläuft ähnliche Change-Prozesse hin zu einer agilen, Tribe-basierten Organisationsstruktur. Ein Tribe ist eine weitgehend autonome Unternehmenseinheit, die für sich profitabel arbeiten muss. Hieraus ergibt sich die Überlegung, wie Wissenstransfer unternehmensweit organisiert und durch Anreize im System gefördert werden kann.

Ziel ist es eine Infrastruktur zu konzipieren, die eine erweiterbare, dauerhafte Wissensbasis aufbaut, die flexibel abrufbar ist. Durch das in der Infrastruktur, an ein On-Demand-Prinzip angelehnte Verfahren, kann die aufwändige Organisation von Schulungen über Tribe-Grenzen hinweg entfallen und das auf der Infrastruktur bereitgestellte Material ist besser auf die Anforderungen in der Domäne angepasst. Hierbei können sich fachliche Ansprechpartner herausbilden und der Austausch über die agilen Einheiten hinweg zielgenauer kanalisiert werden. Allerdings sollte sich dies für die einzelnen Tribes gewinnbringend auswirken. Um Infrastrukturkosten zu reduzieren, bietet sich die Umsetzung eines auf Micropayment basierten Systems an. Dadurch kann gewährleistet werden, dass ein Nutzer größtenteils dafür zahlt, was auch wirklich konsumiert wird. Ein Ansatz, ein solches System zu entwickeln, ist der Einsatz einer Blockchain mit proprietären Tokens. Der Zugriff auf Material aus dieser Inhaltsbasis ist an die Blockchain gebunden und wird über die Tokens freigegeben. Tokens können durch die Firma in Umlauf gebracht und von Tribes bilanziert werden.

Ausgehend von einer Literaturrecherche soll prototypisch ein „Proof-Of-Concept" einer Infrastruktur entwickelt und implementiert werden, welche Wissen als digitales Konsumgut bereit stellt. In der Arbeit werden für diese digitalen Güter vorwiegend Videos betrachtet, um die Verwendung von großen Datenmengen in einem solchen System abzubilden. Daher soll zunächst ermittelt werden, wie das Digital-Rights-Management für die Inhalte auf Basis einer Blockchain und damit die Verwaltung der Berechtigungen zwischen Nutzer und Inhalt funktionieren kann. Weiterhin soll anhand der Endanwendung untersucht werden, ob ein solches System bestehend aus Micropayments und Digital-Rights-Management nutzbar ist.

2 Grundlagen

2.1 Agile Organisationsformen

2.1.1 Strukturen und Mindset

Durch die Digitalisierung und die daraus resultierenden Veränderungen sind klassische Aufbauorganisationen mit deren Unternehmensstrukturen nicht mehr zeitkonform und müssen sich anpassen. Das macht sich vor allem im Arbeitsalltag in kundenorientierten IT-Unternehmen durch die rasante Veränderungsgeschwindigkeit des Marktes bemerkbar. Dadurch wird nun vermehrt auf Agilität gesetzt. Es gibt die verschiedensten agilen Unternehmensformen, sowie Hybriden aus klassischer und agiler Unternehmensstruktur. Diese Hybriden kommen besonders dann zum Einsatz, wenn das Adaptieren der agilen Unternehmensform nur schwer umgesetzt werden kann. Folgende Punkte zeichnen eine agile Organisation aus:

- Selbst organisiert
- Kommunikationsintensiv
- Iterativ
- Adaptiv
- Hierarchiefrei
- Reaktionsschnell
- Markt- und Kundenorientiert

- Innovationsgetrieben

[RUMP19]

Durch neue Innovationen, reaktionsschnelle Markt- und Kundenorientierung, muss folglich, um mit dem Markt mithalten zu können, ein werteorientiertes Arbeiten abgesichert sein. Dadurch ist das aus dem Wasserfallmodell bekannte chronologische, plangetriebene Arbeiten in Frage gestellt und es wird sich nun iterativ orientiert. Dieses iterative Arbeiten, durch beispielsweise zweiwöchentliche Sprints im Scrum Modell, sichert zeitnahes Eingreifen bei Marktveränderungen ab. Daraus resultiert eine kommunikationsintensive Zusammenarbeit, die die genannten Punkte verstärkt. Abschließend muss das System reaktiv sein, um neuen Entwicklungen auf dem Markt schnellstmöglich gerecht zu werden.

2.1.2 Tribe-Modell

Das, vor allem durch Spotify geprägte, Tribe-Modell unterstreicht das agile Mindset. Es ist eine festgelegte Form der agilen Organisation. Die Firma „Spotify AB", bekannt durch ihren Musikstreamingdienst Spotify, hat es eigens für ihr Unternehmen entwickelt und folglich stark geprägt. „Der Streamingdienst Spotify wurde 2006 gegründet, verzeichnet heute Umsätze von über 4 Mrd. Euro und hat 3000 Mitarbeiter. Diese Erfolgsstory kann auf verschiedenste Faktoren zurückgeführt werden. Einer davon ist die agile Organisationsform von Spotify, welche mittlerweile von zahlreichen Unternehmen als Vorbild genommen wird" [LINDNER19]. Auch andere große deutsche Firmen adaptieren das Modell mit dessen Begriffen. So hat die Telekom aktuell etwa 15 Squads in vier Tribes. Außerdem streben auch die ING und Rewe Digital eine Wandel zum Tribe-Modell an. (vgl. [LINDNER19])

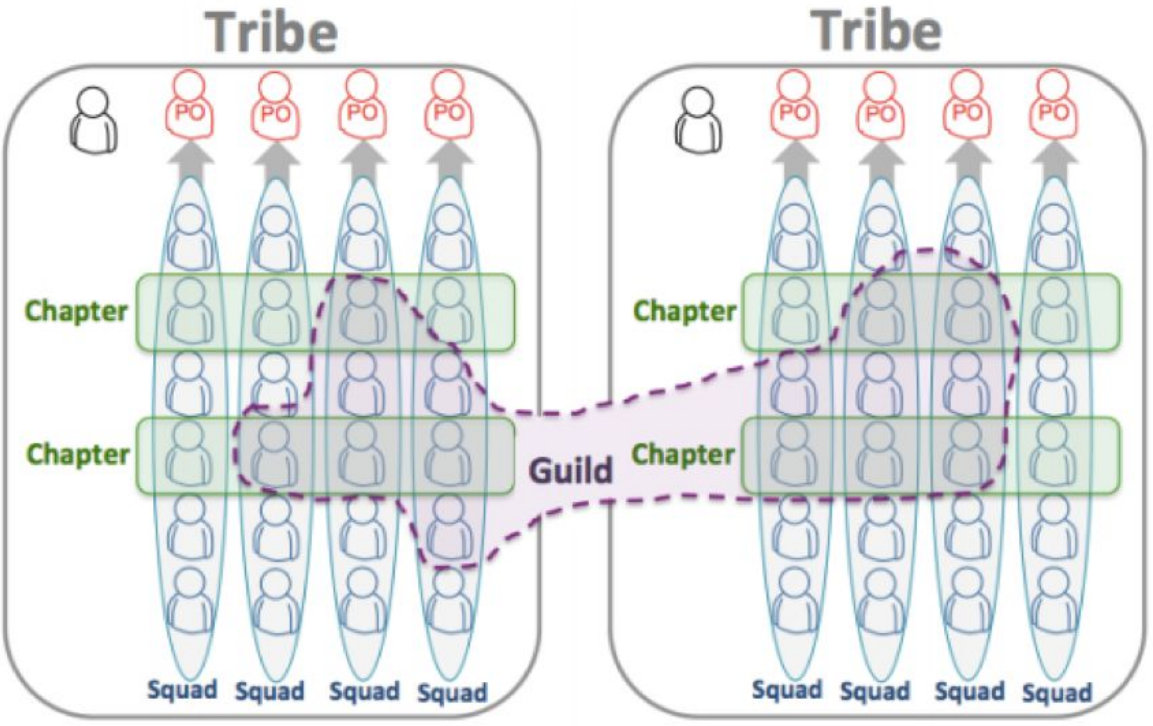

Abbildung 2.1 Tribe-Modell
[LINDNER19]

Abbildung 2.1 zeigt den Aufbau des Modells in seiner Unternehmensstruktur. Man kann es sich zunächst aufgeteilt in einzelne domänen-bezogene Abteilungen vorstellen. Diese werden Tribes genannt. Jeder Tribe hat ein oder mehrere Tribe-Lead, welche meist von den Mitgliedern gewählt werden und sich um organisatorische Aufgaben kümmern oder diese delegieren. Tribes gliedern sich anders als klassische Abteilungen auf. Bei einem Tribe handelt es sich um eine Gruppe von sogenannten Squads, die an der gleichen Domäne arbeiten. Diese müssen für sich selbst wirtschaftlich agieren. Squads sind selbstbestimmte Gruppen mit einem bestimmten projektbezogenen Zweck, die agil arbeiten. Das kann beispielsweise ein Scrum-Team sein, welches meist einem Projekt, einem Produkt, oder einem Teilprodukt zugeteilt ist. Jedes Squad hat einen Squad-Lead, den das Squad festlegt. Wenn Scrum innerhalb des Squads angewendet wird, wird oft der Product Owner oder der Scrum Master dafür festgelegt. In den verschiedenen Tribes kann es sogenannte Chapter geben, diese beschäftigen sich mit einem bestimmten Thema. Der Grund dafür ist, dass dadurch eine fachliche Übereinstimmung abgesichert werden kann. Als Beispiel anzuführen könnten alle Squads in ihrem Projekt Datenbanken verwenden, damit der

Tribe diese einheitlich behandeln kann, gibt es den in den Chaptern abgesicherten fachlichen Austausch. Darüber hinaus gibt es über Tribe-Grenzen hinweg Gilden. Gilden sind verantwortlich für den fachlichen Austausch oder Fortbildungsmaßnahmen für bestimmte Gruppen unternehmensweit, das können zum Beispiel Mobile-Entwickler sein, die sich regelmäßig austauschen, oder zusammen unternehmensweite Aufgaben für diese Spezialfälle lösen. Hier ist beispielhaft anzuführen, dass alle iOS-Entwickler ein unternehmensweiten App Store Auftritt für das Unternehmen pflegen. Das Modell ist von Spotify in weiteren Feinheiten beschrieben, die aber unternehmensabhängig oft angepasst und verändert werden. Die trotz Anpassung gleichbleibende Grundlage der Infrastruktur bildet sich aus Tribes, Chapter und Gilden. (vgl. [WARDT20])

Die Vorteile des Spotify-Modells zeichnen sich eindeutig ab. Trotz der klaren Struktur der Organisationsform ist ein agiles Arbeiten möglich und wird durch die Struktur sogar intensiviert. Dafür sprechen vor allem die Squads, Gilden und Chapter, die in sich flexibel und selbstgesteuert auf ihre Themen fixieren, um möglichst schnell und bestmögliche das Arbeitsziel zu erreichen. Des Weiteren ist durch die Form das Unternehmen schnell skalierbar, was besonders mit dem Wachstumsmarkt in der IT-Branche notwendig ist. Dadurch kann das Eingliedern von neuen Mitarbeitern schnell erfolgen, indem der neue Mitarbeiter in seiner Rolle zu einem Squad, einem Chapter zugeordnet wird und sich gegebenenfalls einer Gilde anschließen kann.

2.1.3 Beispiel adorsys GmbH & Co. KG

Das Unternehmen „adorsys GmbH & Co. KG“ (kurz: adorsys) beschreibt sich als ein innovatives und agiles Softwareunternehmen mit dem Anspruch, dass jeder sich mit seinen eigenen Stärken individuell einbringt. Es bietet sowohl virtuelle Arbeitsplätze als auch Homeoffice an und zeichnet sich durch einen ausgeprägten Teamgedanken aus. Als mittelständischer IT-Dienstleister ist das Unternehmen vor allem in der Bankenbranche tätig. Ähnlich wie viele andere vergleichbare IT-Unternehmen wächst auch adorsys sehr schnell. Mit wachsendem Umsatz, einer immer höheren weltweiten Reichweite und einer steigenden Auftragslage ist es notwendig, um die Produktivität zu skalieren, das Team zu erweitern. Deshalb hat adorsys von 2018 bis hin zum Frühjahr 2020 die Unternehmensgröße von noch familiären 60 auf insgesamt 130 Mitarbeiter gesteigert. Das stellt das Unternehmen vor neue Herausforderungen. 2018

war das Umfeld familiärer und konnte einfacher als ein großes Team fungieren, dass zeichnete sich insbesondere in der Unternehmensführung aus. Um ein Beispiel anzuführen konnten Entscheidungen durch Meetings mit allen Mitarbeitern einfacher getroffen werden. 2020 hingegen kennt nicht jeder den anderen persönlich im Unternehmen und die Koordination wird mit stetig wachsender Mitarbeiterzahl immer komplexer. Beeinflusst durch den innovativen Grundgedanken, hat auch adorsys sich das Tribe-Modell zum Vorbild genommen und organisiert die Firma in Tribes. Dadurch wird für die Firma interessant, welche neuen Möglichkeiten und Herausforderungen das Modell mit sich bringt. Das soll auch Themen, wie schnelles Onboarding, oder auch das Schulen von Mitarbeitern skalierbarer, einfacher und selbständiger bewältigbar machen. (vgl. [ADORSYS20])

Je nach Auftragslage und Nachfrage, werden monatlich neue Mitarbeiter eingestellt, die schnellstmöglich auf ihre Stelle eingearbeitet werden müssen. Durch die Themenvielfalt und die schnellen Veränderung in der IT-Branche, setzt adorsys verstärkt frühzeitig Studenten ein, oft mit geringer Berufserfahrung, für Forschungsarbeiten, Abschlussarbeiten oder Projekten. Diese wiederum müssen durch die meist engen Terminkalender schnell eingearbeitet werden. Vergleichbares gilt für Quereinsteiger oder andere neue Mitarbeiter im Unternehmen. Des Weiteren macht die Möglichkeit von virtuellen Arbeitsplätzen und Homeoffice, Remote-Lösungen notwendig, um auch zuhause domänenspezifisches Wissen für die effiziente Mitarbeit in Teams und Projekten zu konsumieren und weiterzugeben. Ein wichtiger Teil der Domäne als IT-Dienstleister in der Finanzbranche sind sensible Daten, die einen besonderen Umgang voraussetzen. In der Dienstleistungsbranche ist der Umgang mit der Ressource Mitarbeiter, der wohl wichtigste Bestandteil eines intakten Unternehmens. So muss ein Angestellter, je nach Projektlage auch das Projekt wechseln, oder bei erfolgreichen Projektende auf ein neues Projekt umsteigen und in diesem mitwirken. Das verstärkt den Bedarf an domänenspezifischen Wissen.

2.2 Wissenstransfer

2.2.1 Lerntheorie

Beim Wissenstransfer ist die individuelle Leistung des Einzelnen ausschlaggebend. Nur wenn derjenige, der das Wissen weitergibt, es auch vermitteln kann und der Lernende es dann anwenden kann, kann man beim Wissenstransfer von Erfolg sprechen. Es gibt also zwei Seiten, die Vermittlung von eigenem Wissen und das Lernen von neuem Wissen, welches einem beigebracht wird. Beide Seiten betreffen jeden.

Bei Gruppen hängt die mögliche Leistung noch von anderen Faktoren ab. Diese werden in drei Aufgabengebiete gegliedert:

- Additive Aufgaben, bei der die Summe aller individuellen Beiträge...
- Konjunktive Aufgaben, bei der das schwächste Gruppenmitglied...
- Disjunktive Aufgaben, bei der das stärkste Gruppenmitglied...

… die Gruppenleistung ergibt.

Man versucht in der Regel eine Aufgabe so aufzuteilen, dass diese möglichst additiv ist, also jedes Gruppenmitglied den gleichen Beitrag zum Ergebnis beisteuern kann. Das Konzept des „transaktiven Wissenssystems" ist ein Konzept für ein gruppenübergreifendes System und es gliedert sich auf in die Encodierung von Information, in die individuelle Speicherung im Kurz- und Langzeitgedächtnis und damit Nutzung und Vermittlung. In einer Gruppe ist ausschlaggebend, dass soziale Interaktion notwendig ist, da in der Regel jedem einzelnen Mitglied andere Wissensbestände zugänglich sind. Das fordert aber auch, dass die individuellen Mitglieder der Gruppe für Antworten und Fragen Zeit haben. Nach diesem Konzept liegt die Annahme vor, dass Individuen in Gruppen ihre Kompetenzen und Qualifikationen besser nutzen können. Das kommt aber letztendlich auf die konkrete Aufgabenstellung an. (vgl. [NIKODEMUS17, S.54-55])

Drei der bekanntesten Lerntheorien sind Behaviorismus, Kognitivismus und Konstruktivismus. Der Behaviorismus dreht sich um die Betrachtung des menschlichen Verhaltens durch den äußeren Umgang mit Informationen, darunter zum Beispiel

Präsentation, Abfragen und Rückmeldungen. Prinzipiell geht es um das Lernen von reinem Faktenwissen und von den positiven Konsequenzen durch das Verhalten, das zu Wiederholung führen soll. Der kognitivistische Ansatz geht nicht von den äußeren, sondern von dem kognitiven Prozessen des Menschen aus, der die Eigenaktivität und die Motivation des Lernenden in den Mittelpunkt stellt. Lehrende sind hier keine Instruktoren, sondern vielmehr Tutoren, die das Erlernen von Wissen nur anleiten und sich damit die Verantwortung vom Lehrenden zum Lernenden verschiebt. Das individuelle Lösen von Problemen führt zu kognitiven Veränderungen und damit zu langzeitigem Lernerfolgen. Zuletzt der konstruktivistische Ansatz, der vor allem innerhalb der neuen Medien als Lernansatz empfohlen wird. Hier geht es darum, dass Wissen nicht einfach weitergegeben werden kann, sondern durch die individuelle Vorerfahrungen in den eigenen Kontext gesetzt wird. Das entspricht einem aktiven Schaffen der persönlichen Wissensbasis. Die Konstruktion soll gefördert und beschleunigt werden, durch die Diskussion und Reflexion der Lernvorgänge in Gruppen mit einem Moderator, der die Gruppe lenkt. Durch diese Reflexionen wird das Wissen erweitert und verbessert. Alle Lerntheorien setzen jedoch voraus, dass Rahmenbedingungen gesetzt sind, in dem das Individuum lernen und leben kann. So müssen die Projektparameter, sowie die mediendidaktischen und medienpädagogischen Grundlagen lerntheoretisch abgesichert sein. (vgl. [NIKODEMUS17, S.55-58])

2.2.2 Wissenstransfer in der IT

In der Industrie, sowie auch in der IT-Dienstleistung, sind Zeit und Kompetenz wohl die wertvollsten Güter. Es wird viel Zeit in Prozessforschung investiert, um diese perfekt zu optimieren. Als Produktentwickler muss man der Konkurrenz einen Schritt voraus sein, um das Produkt am besten verkaufen zu können. Als Dienstleister muss man den Kunden überzeugen das beste Unternehmen für die Aufgabe zu sein. Deshalb ist insbesondere in der schnelllebigen IT-Branche ein optimierter Wissenstransfer notwendig. Um zum einen durch die Schnelllebigkeit der Technik, diese zu erlernen und verkaufen zu können, und sich an eine spezifische Domäne für den Kunden schnell anzupassen. Durch rasant steigende Komplexität und potenziell explodierenden Nutzerzahlen, müssen diese Prozesse einfach und schnell skalierbar sein. Es kann zu förmlichen Informationsfluten kommen, die geregelt werden müssen. Für diese Flut nutzen Unternehmen oft standardisierte Lösungen. Microsoft hat für

diese Zwecke zum Beispiel OneNote, für schnelle aber auch gut strukturierte Notizen, und Teams zum Remote-Austausch entwickelt. Diese werden weit über das Unternehmen heraus genutzt und finden weltweiten Anklang. Für ein eigenes Wikipedia bietet Confluence eine hervorragende Lösung, um sämtliche Informationen langfristig in eigenen Datenbanken zu speichern. Auch Slack, ein weiteres Kommunikationstool, hat sich gegen die Konkurrenz behauptet und hatte 2019 einen sehr erfolgreichen Börsengang. Natürlich werden auch öffentliche Angebote, wie auf YouTube veröffentlichte Tutorials oder auch offizielle Dokumentationen zu Themen genutzt. Für spezielle auf die Domäne bezogene Informationen, die sich zum Beispiel auf die Sicherheit von Online-Banking beziehen, verlangt es nach eigenen unternehmensinternen Lösungen, Technologien und Architekturen. Denn oft ist die Speicherung auf gemieteten Servern reglementiert.

Wichtig für ein Unternehmen ist, dass die richtige Lerntheorie, beziehungsweise die richtige Mischung aus verschiedenen Theorien passend zu den Mitarbeitern, gefunden wird. Zu beachten ist, dass es sich in Unternehmen meistens um Gruppenleistungen handelt. Das setzt einen guten sozialen Umgang und viel Kommunikation und Reflexion, wie im konstruktivistischen Ansatz erklärt, voraus. Dies kann auch über Remote funktionieren. Scrum setzt zum Beispiel einen festgelegten, regelmäßigen Austausch im Entwicklungsprozess voraus. Allerdings geht es in der IT vor allem beim Einstieg in ein neues Themengebiet oft um Faktenwissen, bei dem meist der behavioristischen Ansatz der mit dem schnellsten Erfolg ist. Zuletzt ist in der Informatik vor allem aber auch das Lösen von komplexen Problemen notwendig, welches langfristig über die kognitive Lerntheorie erlernt wird. Für jede Lerntheorie gibt es Einsatzmöglichkeiten und auch Mischungen sollten in Erwägung gezogen werden.

2.2.3 Studie zum Wissenstransfer in Unternehmen

Nach einer Studie [TANDEMPLOY19] ist der Wissenstransfer in den meisten Unternehmen noch nicht ausgereift. Trotz der allgemeinen Interesse der Mitarbeiter am Wissen der Kollegen teilzuhaben, sowie auch das eigene Wissen zu teilen, werden blockierende Hürden empfunden. Oft gibt es schon Tools und Formate, um Wissenstransfer zu fördern, diese sind aber nur selten im Unternehmen akzeptiert.

Wissen wird öfter durch starre Strukturen als Machtinstrument verwendet und verhindert, dass Mitarbeitende ihrem Drang nach Wissen nachgehen können. Meistens existiert nur ein Austausch mit engen Kollegen, da unternehmensweite Vernetzung nicht gegeben ist. Bei Führungskräften verhält es sich oft anders, diese sind meist über die eigene Abteilung hinaus vernetzt und können oft einfacher ihrem Drang nach Austausch und Vernetzung nachgehen. (vgl. [TANDEMPLOY19])

Dieselbe Umfrage, die für die genannte Studie verwendet wurde, wurde auch bei adorsys durchgeführt. Es haben sich meistens durchschnittlich ähnliche aber zum Teil auch bessere Ergebnisse für funktionierenden Wissenstransfer ergeben. Ein eindeutiger Unterschied zeichnete sich dabei ab, dass Personen mit und ohne Führungsverantwortung, ähnlich geantwortet haben. Ein gutes Zeichen für flache Hierarchien, die in einem agilen Unternehmen Grundvoraussetzung für erfolgreiches Arbeiten sind. Deutlich zu erkennen ist, dass es so wahrgenommen wird als sei zu wenig Zeit vorhanden ist, um sich individuell um Wissenstransfer kümmern zu können. Natürlich füllen priorisiert Kundenprojekte die Arbeitszeit der Mitarbeiter, dennoch sollte Zeit für Wissenstransfer verfügbar sein. Bei der rasant wachsenden Mitarbeiterzahl, die von übergreifend berufseinsteigenden Junioren oder Studenten geprägt ist, ist oft eine individuelle Betreuung von einer Minderheit von Senioren gefordert. Diese ist oft nicht mit einem Tag in der Woche zu decken. Weiterhin, sind durch meist enge Terminkalender, regelmäßige gemeinsame freie Zeiten für Wissenstransfer, schwer zu finden. Das macht den Bedarf an domänenspezifischen Online-Wissensdatenbanken groß.

2.3 Payments

2.3.1 E-Payment-Systeme

Elektronische Zahlungssysteme lassen sich zunächst von traditionellen Zahlungssystemen abgrenzen. Darunter zählt auch Internet-Banking, welches sich lediglich auf die Kommunikation über das Internet und die Vertragsbeziehung zwischen Kunde und Bank beschränkt. Bei einer elektronischen Bezahlung geht es darum, über das gewählte Kommunikationsnetzwerk, zum Beispiel das Internet, eine monetäre Gegenleistung für Güter oder auch Dienstleistungen erbringen zu können.

Systeme, die ein solchen Austausch ermöglichen, werden auch E-Payment-Systeme genannt. Auf der Basis von Software können Urheber diese Systeme aus technischer Sicht beliebig definieren. Man beachte, dass ein E-Payment-System nicht unbedingt im Zusammenhang mit sogenannten E-Geld stehen muss. E-Geld unterscheidet sich von einer beliebigen elektronischen Bezahlung daran, dass es von anderen Unternehmen als der ausgebenden Stelle, sogenannte E-Geld-Institute, als Zahlungsmittel akzeptiert wird. Diese E-Geld-Institute müssen aber zu einer solchen Handlung berechtigt sein und unterstehen den für die Finanzdienstleistungsbranche zuständigen Aufsichtsbehörden ihres jeweiligen Herkunftslandes. (vgl. [DOMBRET08])

Anforderungen an elektronische Bezahlverfahren für sowohl Nutzer als auch Herausgeber sind nach Dannenberg in fünf verschiedene Bereiche eingeteilt:

- Sicherheit
- Benutzerfreundlichkeit
- Geringe Kosten
- Verbreitung
- Flexible Einsatzfähigkeiten

Nur ein sicheres, möglichst schwer angreifbares System, das für die Abhörsicherheit, Authentizität, Datenintegrität, Absicherung im Schadensfall und gegebenenfalls Anonymität für Nutzer und Herausgeber sorgt, ist die Grundlage für ein gut nutzbares Zahlungssystem. Des Weiteren muss es benutzerfreundlich sein, dass alle die das System nutzen, es nutzen können und gern nutzen. Das System darf dem Anwendungsfall entsprechend nicht zu hohe Nutzungskosten fordern und sollte im benötigten Umfeld verbreitbar, sowie flexible einsatzfähig sein. (vgl. [DANNENBERG04], S. 48-51)

2.3.2 Kategorisierung von E-Payments

Die Unterteilung von E-Payment-Systemen ist von Dannenberg nach bestimmten Kriterien klassifizierbar:

- Zeitpunkt der Zahlung
- Höhe der Zahlung

- Transaktionsweg
- Hardware- oder Softwarekomponenten
- Art der Basierung

Mit dem Zeitpunkt der Zahlung, kann man Zahlungssysteme in Pay-before-Modelle, Pay-now-Modelle und Pay-later-Modelle einteilen. Hier steht in Relation, wann eine Belastung des Nutzers erfolgt. Direkt, vorher oder nach dem Erhalt der Gegenleistung. Oft werden Zahlungssystem aber auch an der Größe ihrer Beträge gemessen. Eine Einteilung in Nanopayments ($x<0,05$, $x \triangleq$Betrag), Micropayments ($0,05<x<2,50$), Mediumpayments ($2,50<x<500$) und Macropayments ($500<x$) ist hier üblich. Die Systematisierung nach dem Transaktionsweg unterscheidet zwischen Empfänger- und Absenderinitiierter Zahlung. Beispielsweise ist eine Einzugsermächtigung Empfängerorientiert und eine direkte Zahlung an den Empfänger Absenderinitiiert. Interessant ist die Einteilung in Hardware- oder Softwarekomponenten. Eine Hardwarekomponente wäre zum Beispiel ein Kartenlesegeräte, eine softwarebasierte Lösung kommt rein durch die Software aus. Eine weitere Einteilungsmöglichkeit ist die Basierung eines Zahlungssystems. Beispiele dafür sind Softwarebasierung, Kartenbasierung und viele weitere. Grundsätzlich ist die Kategorisierung eines E-Payment-Systems aber nicht offiziell festgelegt und es kann bei verschiedenen Quellen zu Abweichungen kommen. (vgl. [DANNENBERG04, S. 29-31])

2.3.3 Paid Content-Bereich

Das Internet fordert nach neuen Zahlungssystemen, die vor allem geringe Beträge unterstützen. Der Grund dafür ist, dass im Internet immer weniger Geld durch das Überangebot an kostenlosen Daten verdient wird und Werbeeinnahmen stagnieren. Dadurch wollen Unternehmen vermehrt Geld mit dem Abruf von ihren Daten direkt im sogenannten Paid-Content-Bereich verdienen. Das können Bilder, Texte, Videos und sonstige digitalen Güter sein, die nur wenige Cents kosten sollen. Darüber hinaus gibt es eine starken Drang nach Indvidualisierung. Zum Beispiel hat man früher noch eine ganze CD gekauft, um einen Song zu hören. Heute ist es einfach jeden Song einzeln zu erwerben. Der Kunde kauft nur noch das was er wirklich braucht. Traditionelle Zahlungsmethoden sind dafür nicht geeignet, da sich die Transaktionsgebühren für kleine Beträge für kein solches System eignen. Es müssen winzige Einzeltransaktionen abrechenbar sein. Das ist das Einsatzgebiet für Micropayments. Für den Paid-Content

Bereich gibt es verschiedene Preismodelle. Der Verband Deutscher Zeitschriftenverleger und Sapient benennen Pay-per-Use, Pay-per-Time, Abonnements und Bundles. Für den Micropayment-Bereich sind allerdings nur Pay-per-Use und Pay-per-Time ausschlaggebend. Also die Zahlung für jeden separaten Artikel oder eine Zahlung über eine zeitbezogene Nutzung. Für beide Systeme ist durch die anfänglich kleinen Beträge die Einstiegsschwelle gering. (vgl. [DANNENBERG04, S.68-70])

2.4 Blockchain

2.4.1 Blockchain Grundlagen

„Technisch stellt die Blockchain ("Blockkette") eine dezentrale Datenbank dar, die im Netzwerk auf einer Vielzahl von Rechnern [oft auch Knoten genannt] gespiegelt vorliegt. Sie zeichnet sich dadurch aus, dass ihre Einträge in Blöcken zusammengefasst und gespeichert werden. Durch einen von allen Rechnern verwendeten Konsensmechanismus wird die Authentizität der Datenbankeinträge sichergestellt. Oftmals wird der Überbegriff „Distributed Ledger" als Synonym verwendet, auch wenn nicht jeder Distributed Ledger unbedingt eine Blockkette verwendet" [MITSCHELE18]. Diese Erfindung begründet neue Möglichkeiten der Datenverarbeitung im Internet. Insbesondere durch die Dezentralisierung, durch die massive Datenreplikation, ist eine Datenkorruption bei steigenden Nutzerzahlen praktisch unmöglich. Dies führt zu einer sehr hohen Stabilität eines solchen Netzwerkes gegenüber herkömmlichen Verschlüsselungsverfahren. Darüber hinaus ist Blockchain durch das Peer-to-Peer-Netzwerk als „trustless system" bekannt. Ein Grund dafür ist, dass nur zwei Parteien an einer Transaktion beteiligt sein können. Das bedeutet, dass kein Vermittler, wie zum Beispiel eine Bank oder ein Unternehmen, an der Transaktion beteiligt sein muss. Das bedeutet, dass ein Blockchain-System durch die Dezentralisierung das Risiko von einzelnen Anbietern reduziert und außerdem damit Transaktionsgebühren minimiert, da keine Vermittler und Dritte beteiligt sind. (vgl. [BINANCE20])

Neben den Vorteilen gibt es auch einige Nachteile, die eine Blockchain mit sich bringt. Zunächst sind Daten, die in einer Blockchain hinzugefügt werden, für alle Nutzer

einsehbar und nur sehr schwer modifizierbar. Weiterhin kann die Blockchain, wenn diese zu groß wird, nicht mehr von privaten Nutzern gespeichert werden. Mit sinkender Nutzerzahl, folglich sinkender Knoten Zahl in der Blockchain, sinkt die Sicherheit des Netzwerkes. Außerdem ist das System aufgrund seiner Dezentralisierung nicht für große Daten geeignet, da dies sonst sehr langsam in der Replikation wird. Das Blockchain-basierte Bitcoin Netzwerk braucht mittlerweile über 200GB Speicherplatz und wächst kontinuierlich weiter. Zuletzt kann es durch „51%-Angriffe" zu einer kurzen Übernahme des Systems kommen, soweit das Netzwerk groß ist. Dadurch könnten die letzten Transaktionen modifiziert werden. So ein Angriff ist sehr unwahrscheinlich, da der Angreifer 51% des Netzwerkes übernehmen muss und in der Regel ehrliches Verhalten mit gleichen Ressourcen mehr Profit bringt. Dennoch ist die Blockchain-Technologie durch ihre unvergleichbaren Vorteile einzigartig in ihrer Art und es gibt noch viel ausschöpfbares Potenzial, um Anwendungen zu bauen, die das Potenzial voll nutzen können, sodass es für die Allgemeinheit nutzerfreundlich adaptierbar werden kann. (vgl. [BINANCE20])

2.4.2 Kryptowährungen und Smart Contracts

Blockchain hat sich durch die Kryptowährung Bitcoin etabliert. Die Blockchain-Technologie ist das zugrundeliegende Protokoll über das die Coins, in diesem Fall Bitcoin, sicher übertragen werden können. „Kryptowährungen sind digitale (Quasi-)Währungen mit einem meist dezentralen, stets verteilten und kryptografisch abgesicherten Zahlungssystem" [BENDEL18]. Hier definiert sich auch der grundlegende Unterschied zwischen Tokens und Coins. Coins haben eine eigene unterliegende Blockchain. Tokens hingegen nutzen vorhanden Blockchain-Netzwerke auf die diese als dezentrale Anwendungen implementiert werden können. Beide sind meistens proprietäre Peer-to-Peer-Währungen.

Um einen Coin oder einen Token übertragen zu können empfiehlt es sich ein Wallet, eine digitale Geldbörse, dessen Art durch die Blockchain definiert ist zu verwenden. Die meisten Blockchains funktionieren über Public-Key-Kryptographie. Das bedeutet der Nutzer hat einen privaten Schlüssel mit dem er über sein Wallet auf seinen Blockchain-Account zugreifen kann und darüber verfügen kann. Daneben hat er einen Public-Schlüssel mit dem andere Nutzer des Netzwerkes den Account sehen und ihm

Geld schicken können. Das hat aber den Nachteil, dass alles auf dem Account vorhandene, verloren geht, wenn ein Nutzer seinen privaten Schlüssel verliert.

Wie Abbildung 2.2 zeigt, wird der Wert eines Tokens durch die Entwicklung eines Smart Contracts auf einer vorhanden Blockchain festgelegt. Smart Contracts sind eine Vereinbarung zwischen zwei oder mehr Parteien, die so verbunden ist, dass die korrekte Ausführung durch die Blockchain garantiert wird. (vgl. [CORRALES19]) Das veröffentlichen eines Contracts auf einer Blockchain kostet Coins, welche durch reales Geld erlangt wird. Dieses Geld erhält die „Initial Coin Offering" Stelle der Kryptowährung. Die Investoren erhalten die Tokens. Immer wenn auf der Blockchain etwas vermerkt wird muss eine solche Gebühr bezahlt werden. Dadurch wird abgesichert, dass die Selbstausführung des Smart Contracts funktioniert. Die sogenannten Miner legen die genaue Gebühr fest, da diese die benötigten Ressourcen für den Vermerk auf der Blockchain zur Verfügung stellen.

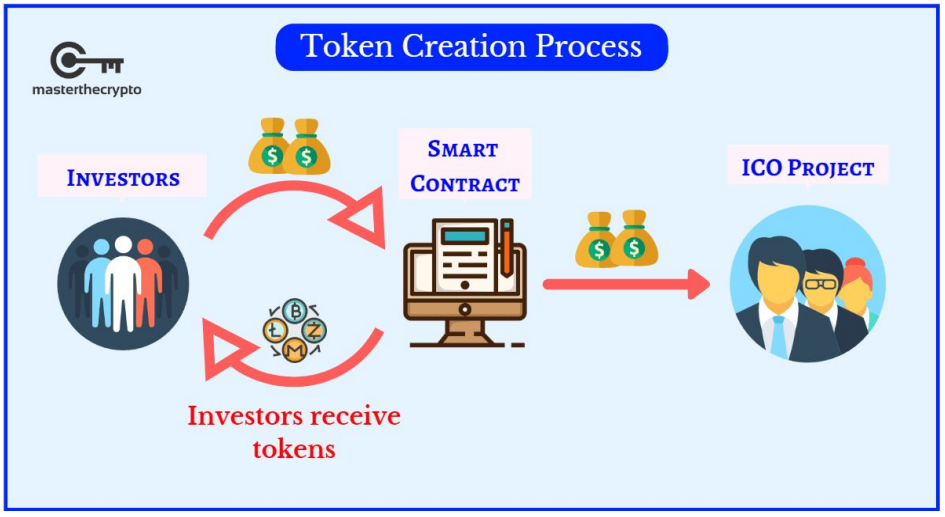

Abbildung 2.2 Token Creation Process
[ZAINUDDIN18]

Immer öfter werden im realen Leben Blockchain-basierte Tokens verwendet. Die Firma Wertgrund hat zusammen mit dem Blockchain-Spezialisten Datarella und dem Münchner Immobilienunternehmen Hammer den „Connex Coin" herausgegeben. Die Macher der Connex-Münze versprechen den Investoren mit dem Token die Möglichkeit, sich schon für 10 Euro an einer Gewerbeimmobilie beteiligen zu können. (vgl. [STREIT20]).

2.5 DRM

2.5.1 Digital-Rights-Management

DRM ist der „Sammelbegriff für alle technischen Maßnahmen zur digitalen Kontrolle von Urheber- bzw. Verwertungsrechten an Content aller Art (Urheberrecht, Verwertungsrecht). Grundprinzip ist die Markierung und/ oder Verschlüsselung digitaler Inhalte mit der Konsequenz der Einschränkung von Nutzung und Weitergabe. Die Markierung erfolgt durch sog. digitale Wasserzeichen, die sichtbar oder unsichtbar sein können. Die Verschlüsselung geht mit einer Chiffrierung einher, die nur mit einem passenden „Schlüssel" überwunden werden kann. Entsprechende Schlüssel können soft- oder hardwarebasiert sein. Systeme für den digitalen Rechteschutz (DRM-Systeme) verfügen im Allgemeinen über vier grundlegende Funktionsbereiche: Zugangssteuerung, Nutzungssteuerung, Abrechnung sowie Verfolgung von Rechtsverletzungen. DRM-Systeme bieten den großen Vorteil, den Schutz geistigen Eigentums im Internet zu erleichtern. Gleichzeitig führen sie durch ihren teilweise sehr restriktiven Charakter aber auch zu verminderter Nutzung entsprechend geschützter Inhalte" [LACKES18].

Grundsätzlich geht es bei DRM darum, dass bestimmten Content nur Berechtigte sehen können. In der Regel heißt das, dass bei Übertragungen im Internet der Content meist verschlüsselt ist. Entschlüsseln können nur der Urheber und die Berechtigten. Das zentrale Problem bei DRM ist, dass es immer Wege gibt, um Verschlüsselungsverfahren auszutricksen oder zu umgehen. In dieser Arbeit wird aber davon ausgegangene, dass die genannten Verschlüsselungsverfahren sicher sind. Zwei bekannte Verschlüsselungsverfahren sind symmetrische und asymmetrische Verschlüsselung. Bei symmetrischer Verschlüsselung wird der Content mit dem selben Schlüssel verschlüsselt wie auch entschlüsselt. Bei asymmetrischer Verschlüsselung haben beide der Urheber und der Empfänger einen öffentlichen Schlüssel und einen privaten Schlüssel. Der Urheber kann mit dem öffentlichen Schlüssel des Empfängers den Content verschlüsseln und dem Empfänger schicken. Der Empfänger kann den Content mit seinem privaten Schlüssel entschlüsseln.

2.5.2 Blockchain als Teil von DRM

Um Blockchain über ein Zahlungs-Konzept hinaus zu verwenden und in ein richtiges DRM-System zu adaptieren, gibt es noch keine offiziellen Konzepte. In den letzten Jahren wurde Blockchain aber schon einige Male als fortschrittliches Sicherheitskonzept verwendet. Ein jordanische Flüchtlingslager versucht, Menschen ohne staatliche Ausweispapiere oder ein Bankkonto in ein Finanz- und Rechtssystem zu bringen. Durch die Abtastung über die Iris mit einer Kamera wird die Identität der Lagerbewohner über eine Ethereum ähnliche Blockchain authentifiziert und dadurch können sie bezahlen. (vgl. [JUSKALIAN18]) Selbst Unternehmen wie Sony investieren viele Ressourcen in die Blockchainforschung. Sony hat ein spezielles Rechteverwaltungs-Konzept patentiert, bei dem die Rechte in einer Blockchain gespeichert werden. Vielmehr als die Architektur in Abbildung 2.3. verrät Sony aber nicht.

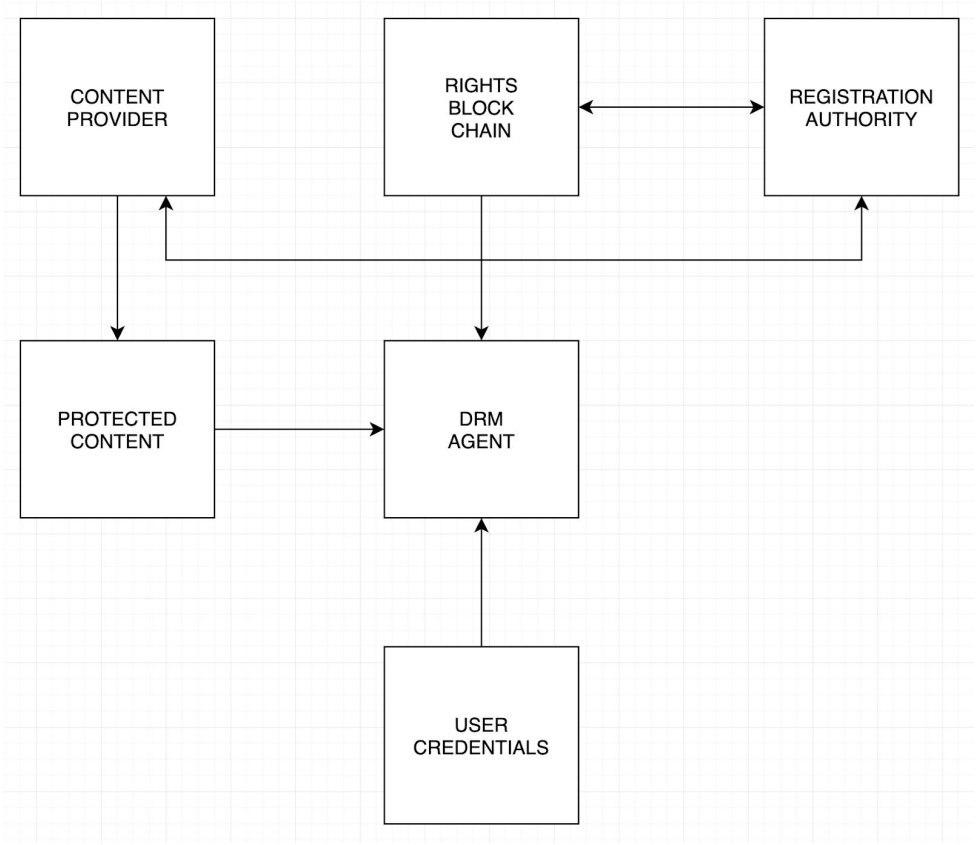

Abbildung 2.3 Blockchain Rechteverwaltung Sony
(vgl. [SONYGROUP18])

3 Konzeption Infrastruktur

3.1 Infrastrukturelle Rollen

Ausgehend von einer Tribe-Organisation wird in dieser Arbeit eine Infrastruktur zum domänenspezifischen Wissenstransfer entwickelt. Diese Infrastruktur soll eine erweiterbare, dauerhafte Wissensbasis auf der Grundlage von Videos schaffen, die flexibel abrufbar ist. Im Rahmen der Forschung soll ein Blockchain-basiertes System zum Einsatz kommen mit dem das Digital-Rights-Management organisiert wird. Um unternehmensseitig die Infrastruktur zu fördern, sollen als Anreizsystem für die Nutzer Tokens durch Micropayments fungieren, die auf der Blockchain aufgesetzt sind. In der Unternehmensstruktur sind Tribes Einheiten, die für sich selbst profitabel arbeiten. Dadurch hat jeder Tribe durch Selbstorganisation eine Mitverantwortung für die Kompetenz seiner Mitglieder. Somit ist jeder Tribe eine autonome Einheit und erhält in der Infrastruktur deshalb ein Token-Account, das den Zugriff auf die Blockchain gewährt, um darüber durch Tokens Inhalte konsumieren zu können. Im folgenden wird der Austausch von Wissen in Form von Videos betrachtet. Diese digitalen Inhalte können gegebenenfalls mit wenigen Anpassungen der Infrastruktur durch andere Arten von Content getauscht werden.

Die Infrastruktur bildet ein System zur strategischen Organisation von Wissen. Dieses System ist für diese Arbeit in verschiedene Rollen-Bezeichnungen gegliedert. Im Folgenden sind die wichtigsten Bezeichnungen erläutert:

- **Infrastruktur**: Softwarearchitektur zur strategischen Organisation von Wissen.
- **System**: Die Infrastruktur, die beteiligte Hardware und Peripherie.
- **Organisation**: Bereitsteller des Systems.
- **Nutzer**: Alle individuellen Nutzer des Systems, hauptsächlich die Angestellten der Organisation.
- **Clients**: Die Endgeräte über welche die Nutzer auf das System zugreifen können.

- **Konsument (Consumer)-Client**: Der Client über den Videos konsumiert werden können.
- **Produzent (Producer)-Client**: Der Client über den produzierte Videos hochgeladen werden können.
- **DRM (Digital-Rights-Management), Management-System, Management-Server**: Die serverseitige Implementierung zur Verwaltung der Daten.
- **Ethereum**: Die verwendete Blockchain.
- **Smart Contract**: Der Vertrag über den die Tokens auf der Blockchain bereitgestellt werden.
- **Token**: Eine Einheit einer über den Smart Contract nutzbaren Währung.
- **Token-System**: Die durch den Smart Contract festgelegten Eigenschaften der Tokens.

3.2 Technisches Grundkonzept

Jeder Tribe soll zum einen Videos produzieren und zum anderen konsumieren können. Folglich werden bis zu zwei verschiedene Arten von Clients benötigt. Einen Client für den Upload von Videos und einen für das Konsumieren von Videos oder einen der beides kann. Zwischen den Clients ist im Netzwerk ein DRM über den der Austausch erfolgen kann. Dieses Managementsystem verwaltet die generierten Daten in Datenbanken und ist über eine Blockchain an einen Smart Contract angebunden. Über diesen Smart Contract werden Tokens initialisiert und verteilt. Dadurch kann das Managment-System die Tokens zwischen Clients verwalten. Die Tokens sind Micropayment-basiert und sorgen für eine Bilanzierbarkeit im gesamten Systems. In Abbildung 3.1 ist die Infrastruktur mit ihren Kommunikationswegen zusammengefasst. Der folgende Text bezieht sich auf diese Abbildung und fasst das technische Konzept zusammen. In Kapitel 3.4 bis 3.7 werden die Zusammenhänge genauer erläutert.

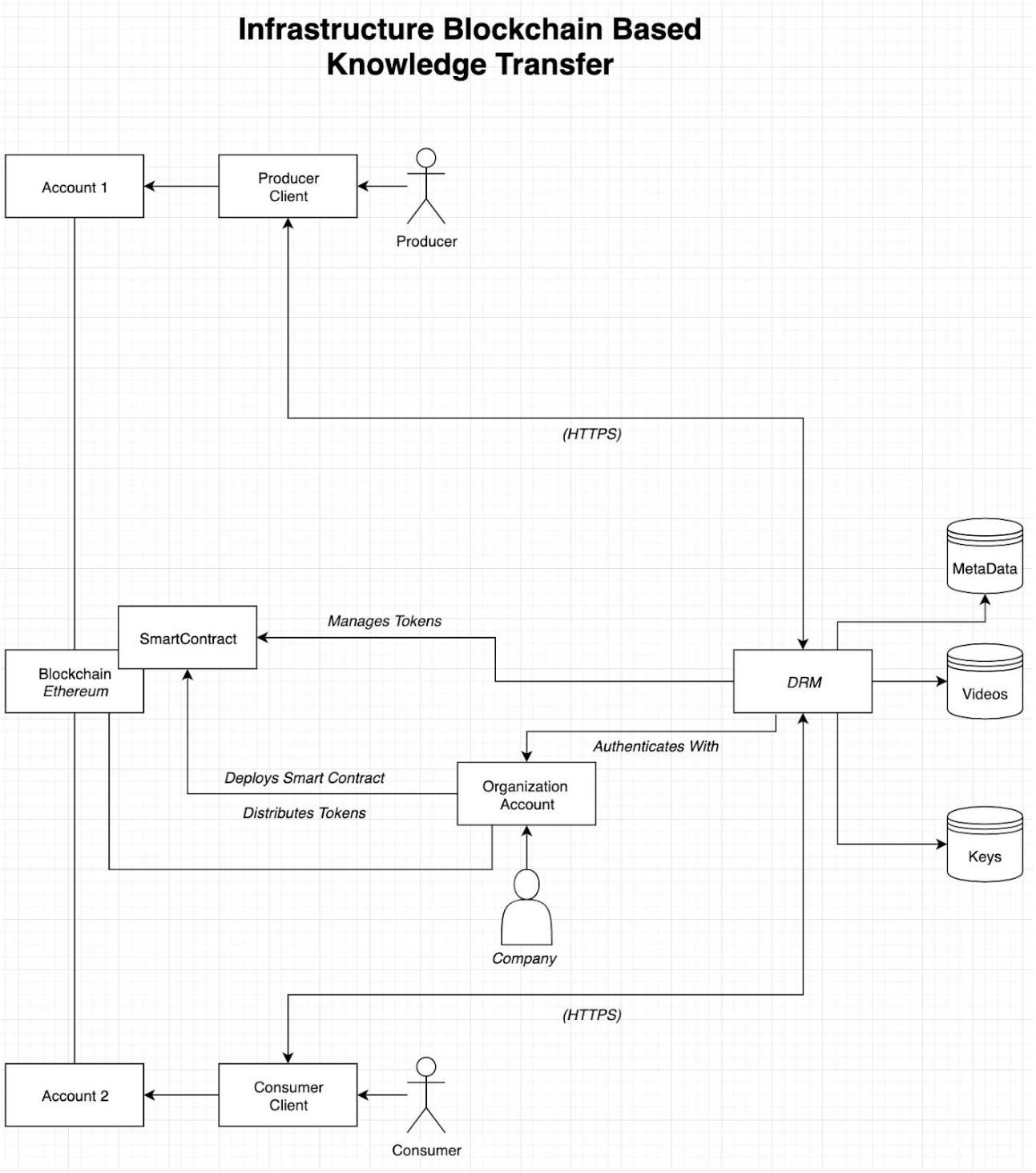

Abbildung 3.1 Konzeptentwurf Infrastruktur Blockchain-basierter Wissenstransfer

Beginnend mit dem Producer kann dieser Videos in das System laden. Für den Upload über den Producer-Client werden diese symmetrisch verschlüsselt und werden auf das DRM hochgeladen. Durch dem Producer seine öffentliche Account-Adresse wird dieser

dort als Urheber gespeichert. Des Weiteren schickt der Producer den Schlüssel vom verschlüsselten Video gesichert an den Management-Server. Dieser speichert Schlüssel, Account-Adresse und Videos sicher in einer Datenbank. Will nun ein Nutzer ein Video anschauen nutzt dieser den Consumer-Client zum Streamen des Videos und schickt einen Request für das entsprechende Video an das Management-System. Über den Request gibt dieser seine Account-Adresse mit. Das Management-System überprüft, ob der Client genügend Tokens in seinem Account hat und gibt gegebenenfalls den Stream frei, nach den Prinzipien von Pay-per-Use oder Pay-per-Time sendet das Managment-System Tokens vom Account des Konsumierenden zum Account des Urhebers. Die Tokens sind durch einen Smart Contract auf eine bestimmt Blockchain durch die Tribe-Organisation aufgesetzt worden und wurden an die Tribes verteilt. Am Ende einer von der Organisation festgelegten Zeitspanne können genutzte Tokens in eine reale Währung umgewandelt und bilanziert werden. Wie viele Tokens verteilt werden und wie viel ein Token wert ist, muss das Unternehmen für sich selbst bestimmen.

3.3 Werte - Value Proposition Canvas

Durch das Value Proposition Canvas in Abbildung 3.2 werden alle Werte für die Nutzer der Infrastruktur analysiert. Die Zielgruppe der Infrastruktur teilen sich in die Organisation und in deren Angestellten auf. Es werden drei Bereiche mit deren Herausforderungen und deren Lösungen behandelt. Diese teilen sich auf in Nutzen, Aufgaben und Herausforderungen für alle Nutzer der Infrastruktur. Um diese Ergebnisse zu ermitteln wurde unter anderem auf Basis der Erfahrung von adorsys gearbeitet. In diesem Abschnitt werden die grundsätzlichen Aufgaben des Systems geklärt. Die Auseinandersetzung mit ähnlichen Systemen und Vor- und Nachteilen findet im späteren Verlauf der Arbeit statt (siehe Kapitel 5).

Value Proposition Canvas

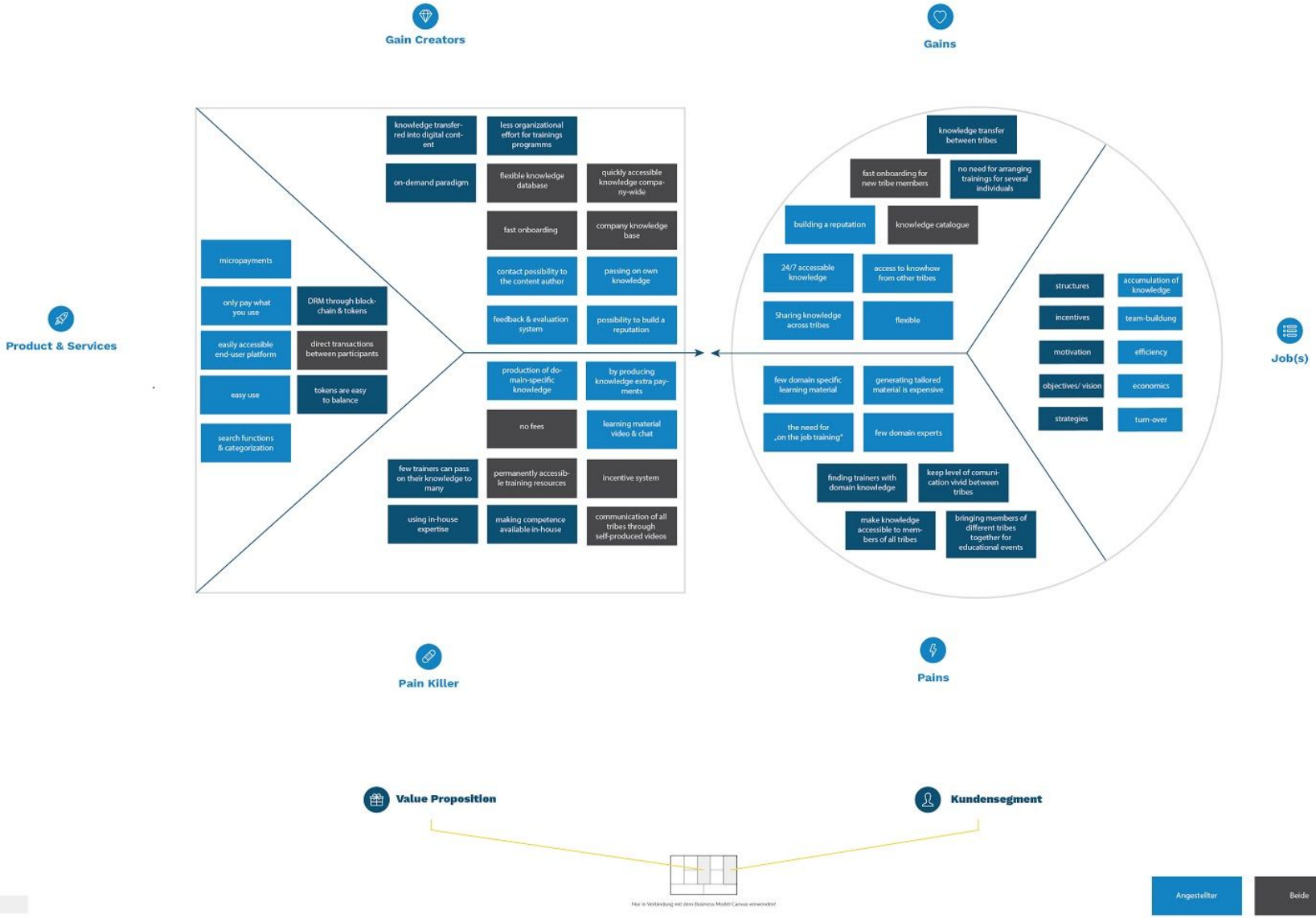

Abbildung 3.2 Value Proposition Canvas

3.3.1 Kunden-Job(s)

Welche grundlegenden Aufgaben aller Art kann das System für den Einsatzbereich in der Firma lösen? Die Angestellten möchten möglichst effizient und wirtschaftlich arbeiten können und müssen dabei mit ihrem Tribe einen bestimmten Deckungsbeitrag erwirtschaften. Darüber hinaus wollen sie Wissen aufbauen, teilen und ihre Teams und Firma sozial stärken. Die Firma möchte die Angestellten bei ihren Aufgaben unterstützen, motivieren und die besten Anreize schaffen diese zu erledigen. Dabei sollen bestimmte Ziele der Firma möglichst strukturiert verfolgt werden und durch zielführende Strategien organisiert sein.

Durch die Blockchain mit ihren im vorherigen Kapitel beschriebenen Vorteilen, dem DRM-Management und dem aufgesetzten Token-System soll dies möglich sein. Ein gut aufgebautes Rechtemanagement schützt die Informationen der Firma und die Nutzung des Systems ist über Tokens strukturiert überschaubar. Durch sogenannte „Learning Paths", die eine chronologische Abfolge von Videos zu einem Thema darstellen, kann ein Angestellter in strukturierter Form auch komplexere Lernziele erstreben. Aber auch das individuelle Nachschauen von Informationen wird durch das Micropayments-System gefördert. Nur was man auch wirklich nutzt muss man bezahlen. So kann ein Angestellter sich nur für einen Ausschnitt eines Videos interessieren und diesen konsumieren. Durch Suchfunktionen, Kategorisierungen, Bewertungssysteme und Kapitel-Sprungpunkten in den clientseitigen Endanwender-Apps kann dies darüber hinaus gefördert werden. Das System ist für Angestellte durch Apps, die für das persönliche Endgerät kompatibel sind, leicht erreichbar und nutzerfreundlich gestaltet. Durch direkte Transaktionen unter den Tribes wird darüber hinaus unternehmensweiter Austausch gefördert.

3.3.2 Herausforderungen

Die Vorteile des Systems kommen aber erst durch die Lösung von speziellen Herausforderungen für sowohl das Unternehmen als auch für die Angestellten im Arbeitsalltag zum Vorschein. Oft ist das Finden von guten Trainern nicht leicht, vor allem wenn es um domänenspezifisches Wissen geht, dass besonders bei der Ausbildung am Arbeitsplatz gebraucht wird. Diese Trainer können unternehmensintern

oder extern sein. Weiterbildungsmaßnahmen während dem aktiven Arbeiten an einem Projekt durch sich ändernde Umstände, macht vor allem in der schnelllebigen IT-Branche „On-The-Job-Training" sehr wichtig. Dafür ist projektbezogenes Lernmaterial notwendig. Um ein Experte zu sein, muss dieser die Domäne kennen und Unterrichten können. Das fordert oft viele Jahre Berufserfahrung auf dem Gebiet und darüber hinaus. Domänenspezifisches Wissen kann oft nicht einfach durch öffentliche Quellen erlangt werden, sondern benötigt speziellen Zugang zu Informationen. Dieses Informationsmaterial zu beschaffen ist meist sehr teuer, da die wenigen Experten durch die spezielle Expertise sehr teuer sein können. Das blockiert effektive Arbeitsflüsse im Unternehmen für beide, die Angestellten und das Unternehmen an sich. Des Weiteren funktioniert das Zusammenbringen von Mitgliedern verschiedener Tribes für Schulungsveranstaltungen, aufgrund von abweichenden Terminkalendern nur in unregelmäßigen Abständen. Ebenso ist es für den internen Austausch notwendig eine regelmäßige Kommunikation unternehmensweit aufrecht zu erhalten.

Die Infrastruktur des Systems ist darauf ausgelegt, die Produktion von domänenspezifischen Wissen zu fördern und damit eine dauerhafte, erweiterbare Wissensbasis zu schaffen. Die Förderung fungiert über das Token-Anreizsystem und läuft durch Micropayments kosteneffizient durch annähernd keine Gebühren. Dadurch können wenige Trainer ihr Wissen an alle im Unternehmen weitergeben, die gesamte Kompetenz des Unternehmens gesteigert und nutzbar gemacht werden und die Kommunikation unter den Tribes wird durch die Infrastruktur selbst verbessert. Außerdem wird das Mindset der Angestellten positiv gefördert, da sie nun Wissenstransfer offiziell bilanzierbar nutzen können und sie die aufgebrachte Zeit als produktiv empfinden.

3.3.3 Mehrwert

Was für einen Mehrwert bringt das System für die potentiellen Nutzer? Die Organisation möchte, dass Wissenstransfer unternehmensweit zwischen den verschiedenen Tribes stattfinden kann. Darüber hinaus möchte sie keine sich wiederholenden Schulungen für viele verschiedene einzelne Angestellten organisieren. Es gibt Schulungen, die aufgrund von individueller Betreuung nur eine geringe Anzahl von Teilnehmern zulassen und öfter durchgeführt werden müssten. Außerdem möchten alle, sowohl die Organisation als auch die Angestellten, dass neue

Angestellten, die der Organisationsform geschuldet alle einem Tribe beitreten müssen, für den Einsatz zeitnah angelernt werden können. Des Weiteren wäre für beide ein unternehmensweiter Wissenskatalog ein Gewinn. Die Organisation könnte sich darüber repräsentieren und die Angestellten könnten das Wissen nutzen, um effektiver zu arbeiten. Ein weiterer Nutzen für die Angestellten wäre, eine persönliche Reputation aufzubauen, individuell oder für den Tribe, um sich darüber zu repräsentieren. Ein Wunsch wäre auf unternehmensweites, domänenspezifisches Wissen rund um die Uhr flexibel zugreifen zu können ohne das Tribe-Grenzen ein Hindernis darstellen.

Durch die Infrastruktur sollen diese Wünsche gedeckt werden. Durch die On-Demand erreichbaren Videos kann eine unternehmensweite Wissensbasis und individuelle Reputationen geschaffen werden. Diese Wissensbasis ist immer schnell erreichbar und ist flexibel skalier- und selektierbar. Außerdem kann man sein eigenes Wissen durch das Token-System unmittelbar bilanzierbar weitergeben. Die Selektion kann über Playlisten, oder Learning Paths stattfinden. So kann Onboarding schnell funktionieren, indem man dem Neuling, die zu bewältigenden Learning Paths gibt und er diese absolvieren kann. Aber natürlich ist individuelles Feedback notwendig und Fragen können auftauchen. So ist immer eine Kontaktmöglichkeit zum Content Autor oder zu Wissensträgern gegeben, die der Lernende über Kommunikationsplattformen erreichen und gegebenenfalls ein persönliches Gespräch organisieren kann. Der individuelle Nutzer soll für den effizienten Nutzen für Nachfolger die einzelne Videos bewerten und Feedback über ein Ratingsystem oder Kommentarsystem geben können. Diese müssen gemonitort werden. Auf Dauer spart das System dem Unternehmen organisatorischen Aufwand und betreibt strategischen Wissensaufbau. Durch die Wissensbasis ist das Unternehmen unabhängiger von individuellen Angestellten und festigt damit die Struktur dauerhaft. So kann das System den Schaden, der durch den Ausfall oder Wechsel von Angestellten verursacht wird, mindern.

3.4 Token-Management

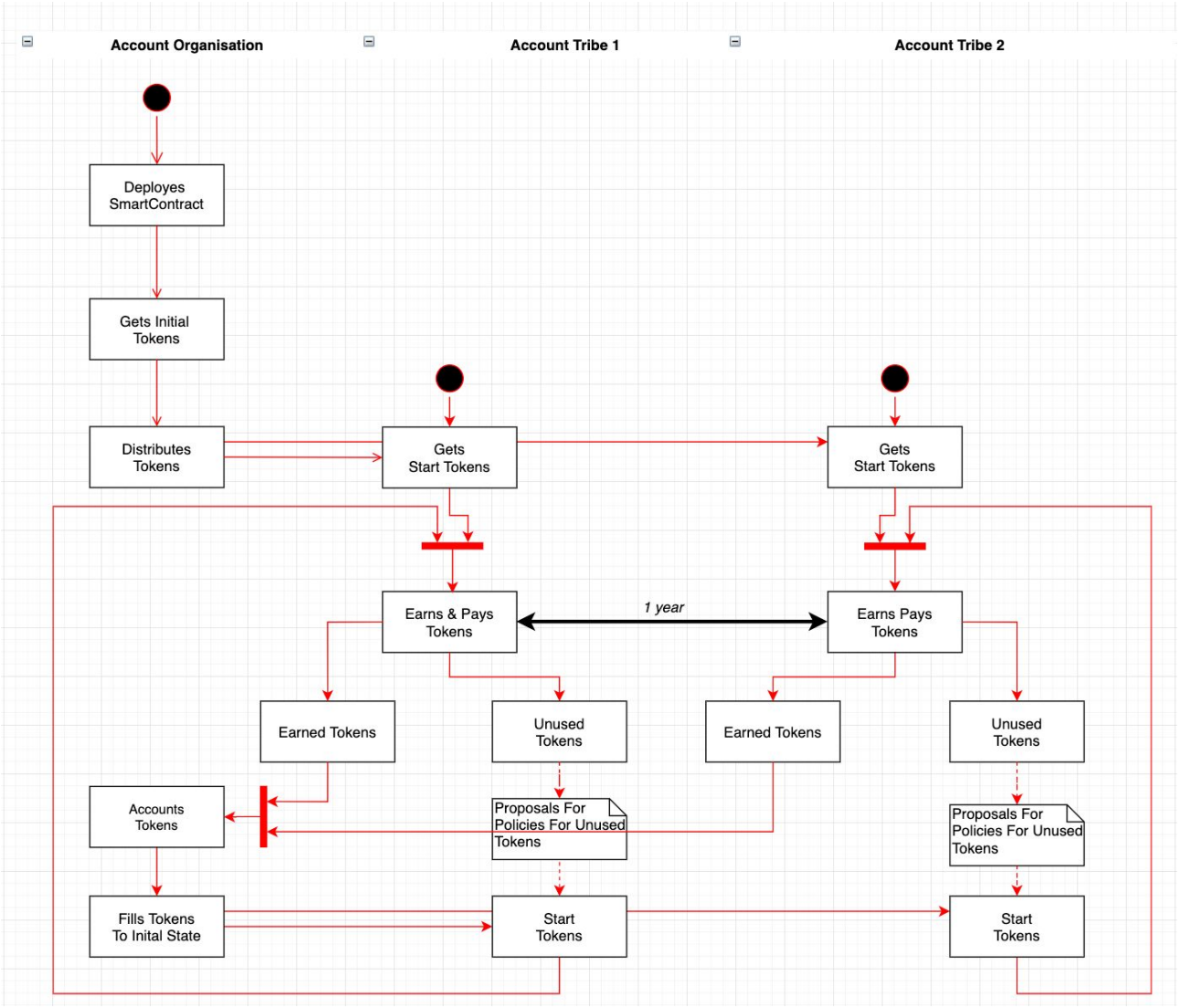

Abbildung 3.3 Token-Management

Es wird mit eigenen proprietäre Tokens gearbeitet, die nur im eigenen Unternehmen funktionieren und damit die domänenspezifischen Daten absichern. Das ist auch der Grund dafür warum keine Coins verwendet werden, da diese direkt an die öffentliche Blockchain gekoppelt sind und auch darüber hinaus verwendet werden können. Durch einen auf einer Blockchain basierenden Smart Contract, können solche proprietäre Tokens initialisiert werden. Die Eigenschaften der Tokens können je nach individualisierten Smart Contract an das spezifische Umfeld des Systems angepasst werden. Der folgende Prozess ist in Abbildung 3.3 beschrieben und wird im folgenden

Text erläutert. Die Tokens werden zunächst in den Account des Smart Contract Deployers bereitgestellt. Dieser fungiert nun als Ausgabestelle, meist übernimmt diesen Schritt das Unternehmen. Danach kann durch die Ausgabestelle eine Verteilung der Tokens auf die Accounts der verschiedenen Nutzungsstellen erfolgen. Im Tribe-Modell sollte idealerweise eine gerechte Verteilung auf die einzelnen Tribes nach ermessen des Unternehmens erfolgen. Nun können die Tribes über deren Selbstorganisation die Tokens verwenden, um Videos zu konsumieren und durch Produktion Token verdienen. Der Konsum und die Produktion erfolgen über ein bestimmte von der Ausgabestelle vorgegeben Zeitspanne. Dabei kann es sich beispielsweise um ein Geschäftsjahr oder ein Quartal handeln. Sobald das Jahr vorüber ist, wird die Bilanz gezogen und die Tokens, die ein Tribe verdient hat, werden umgerechnet und dem Tribe als Umsatz angerechnet. Die ungenutzten Tokens werden nicht umgerechnet. Dadurch wird verhindert, dass ein Tribe ohne das System zu nutzen von dem System profitieren kann. Für die ungenutzten Tokens sollte nach Unternehmensstandard Regeln festgelegt werden wie damit umzugehen ist. Nach erfolgreicher Bilanzierung, werden die Tokens wieder aufgefüllt und der nächste Zyklus beginnt. Im Tribe-Modell muss ein Deckungsbeitrag von jedem Tribe erwirtschaftet werden. Je nach wirtschaftlicher Kraft des Unternehmens muss die Verteilung und der Wert der Tokens mit dem Deckungsbeitrag dafür möglichst profitabel errechnet werden.

3.5 Produktion von Videos

Um das System zu nutzen, müssen Daten produziert werden. Anhand einer Video basierten Lösung wird das in dieser Arbeit erarbeitet. Eine Nutzungsempfehlung wäre, dass mindestens ein Aufnahme Arbeitsplatz eingerichtet wird. Das kann sehr einfach ausfallen, indem eine Kamera und ein Mikrofon vor einer Tafel steht und man gegebenenfalls noch Präsentationen etc. zuschalten kann. Je nach Nutzungsgrad und Nachfrage kann man dies nach Bedarf und Vorlieben gestalten und betreuen lassen. Aber die Produktion von Videos kann auch ganz einfach über den Computer des Nutzers funktionieren, beispielsweise mit kommentierten Bildschirmaufnahmen, gegebenenfalls noch einer FaceCam. Auch hier sollte nach Bedarf entsprechende Software zur Verfügung gestellt werden. Über einen Ansprechpartner kann man sich

einweisen lassen und nachfragen. Dies sind nur Empfehlungen, wie eine solche Videoproduktion funktionieren kann.

3.6 Daten-Management

3.6.1 Clients

Ausgehend von zwei unterschiedlichen Clients, einem zum Hochladen und einem zum Konsumieren der Videos, müssen sich die Nutzer in beiden Clients zu ihrem Tribe mit dem Tribe-Account authentifizieren. Dies geschieht über die Eingabe oder Auswahl der öffentlichen Account-Adresse des Tribes, welche vom Nutzer über den privaten Schlüssels des Accounts bestätigt werden kann. Um diesen Schritt zu vereinfachen kann dies auch über eine QR-Code-Scan funktionieren, der die Schlüssel beinhaltet. Die Clients können mit dem Management-Server sicher kommunizieren. Die sichere Übertragung kann beispielsweise über eine HTTPS-Verbindung oder eine asymmetrische Verschlüsselung erfolgen. Im folgenden wird bei jeder Übertragung von einer potentiell sicheren HTTPS-Verbindung ausgegangen. Die Verschlüsselungen von großen Dateien werden im Client und nicht im Server behandelt, um bei diesen ressourcenlastigen Vorgang Ressourcen im Server zu sparen. Bei steigender Nutzerzahl würde der Server sonst linear steigernd mehr Rechenleistung brauchen. Die meist rechenstarken Clients können eine Datei einfacher verschlüsseln und diese somit auch sicher an den Server übertragen. Eine Empfehlung wäre ein solches Management-System ohne überdurchschnittliche Ressourcen nicht selbst zu entwerfen, da es schon sehr gute Lösungen von großen Softwareherstellern gibt. So hat Apple, für seine eigenen Geräte „Apple-Fairplay” als DRM für HTTP-Live-Streaming entworfen, welches über Lizenzen erwerbbar und konfigurierbar für den kommerziellen Nutzen ist. Mit Apple-Fairplay kann nicht als Video DRM-Plattform gearbeitet werden, da sich der Prozess als Entwickler zu registrieren als komplex herausstellte und damit für ein „Proof-of-Concept” zu umfassend wäre. Dieser komplexe Prozess ist für Apple notwendig, damit sie ihr eigenes DRM-Sicherheitssystem schützen können und die Implementierung nicht öffentlich für potentielle Angriffe zur Schau stellt. Auch Google hat mit seinem eigenen System „Widevine” ein vergleichbares System geschaffen. Da diese Arbeit aber

„Proof-Of-Concept" ist, wird im Folgenden eine vereinfachte Form eines eigenen DRMs in der Theorie betrachtet.

3.6.2 Producer-Client

Der folgenden Prozess ist im Aktivitätsdiagramm in Abbildung 3.4 beschrieben. Ist ein Video produziert, kann der Producer das Video über den Upload-Client hochladen. Zunächst muss der Produzent das Video in den Client laden und den Upload freigeben. Mit selbst generierten symmetrischen Schlüsseln, wird das Video beziehungsweise einzelnen Datenpakete, die zum Streamen notwendig sind, im Client verschlüsselt und die verschlüsselten Pakete werden in das Managment-System hochgeladen. Außerdem übergibt der Producer seine öffentliche Account-Adresse, die symmetrischen Schlüssel und Metadaten zu dem Video an den Management-Server. Die Metadaten können über das Interface im Client über Eingabefelder mit Titel, Beschreibung und sonstigen Informationen erweitert werden. Die verschlüsselten Dateien werden in einem Dateisystem gespeichert. Die anderen Daten können in einer klassischen Datenbank gesichert sein. Damit ist der Upload-Vorgang abgeschlossen.

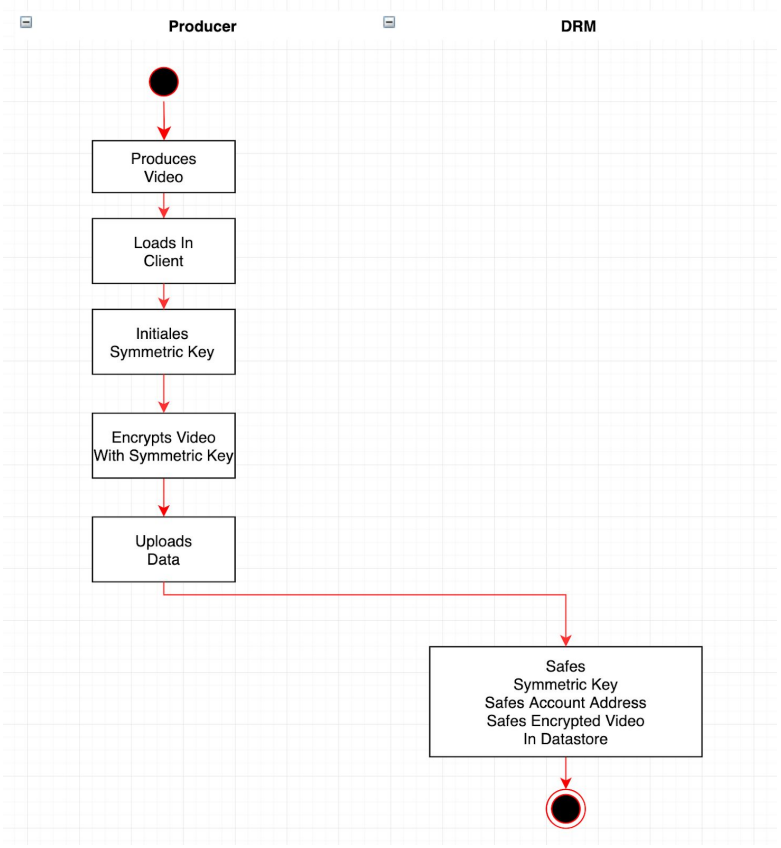

Abbildung 3.4 Prozess Upload-Client

3.6.3 Consumer-Client

Der Konsum eines Videos erfolgt über den Consumer-Client, wie im Prozess in Abbildung 3.5 beschrieben und im folgenden näher erläutert. Ein Nutzer, der ein Video konsumieren möchte, öffnet den Consumer-Client und kann dort über die vom Management-Server bereitgestellten Metadaten der Videos sich über die entsprechenden Videos informieren und dasjenige heraussuchen, das er betrachten möchte. Hat er sich für eins entschieden, kann er das entsprechende Video bestätigen und über den Client wird ein Request an den Management-Server geschickt. Mit diesem Request authentifiziert sich der Client mit dem öffentlichen Schlüssels des Tribe-Accounts und muss eine bestimmte Anzahl von Tokens, für die Überweisung über den Smart Contract dem Management-System freigeben. Das Management-System überprüft über den Smart Contract, ob der Nutzer genügend Tokens zur Verfügung hat und wie viele zur Überweisung freigegeben sind. Ist das nicht der Fall, wird die Anfrage abgelehnt und zurück an den Client geschickt. Bei erfolgreicher Token-Freigabe startet das Management-System einen Videostream und schickt den jeweiligen Schlüssel über einen separaten Request mit dem das Datenpaket entschlüsselt werden kann. Solange der Nutzer Tokens hat, werden diese über Pay-per-Use oder Pay-per-Time vom DRM abgebucht. Sobald keine zu überweisenden Tokens mehr vorhanden sind, wird der Stream abgebrochen. Der Prozess, dass keine Tokens vorhanden sind kann individuell behandelt werden. Ein Beispiel wäre im Client zu empfehlen weitere Tokens freizugeben oder sich neue zu erwerben. Aber das System kann den Endgerät-Nutzer auch nach einer bestimmten Zahl für Requests sperren, um beispielsweise Brute-Force Angriffen zu trotzen.

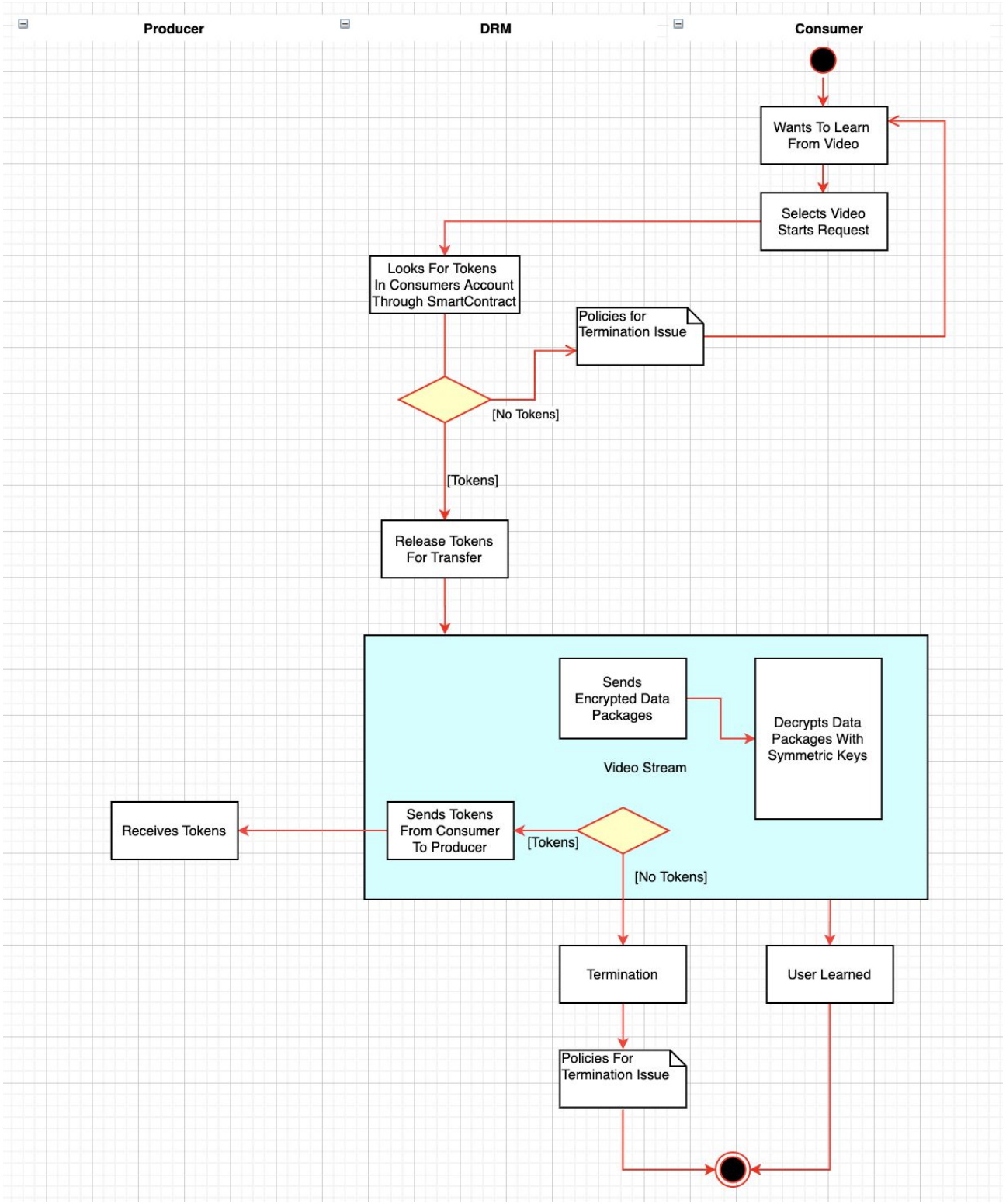

Abbildung 3.5 Prozess Consumer-Client

3.7 Interaktion von Tokens, Daten, Nutzern

Angetrieben durch das Token-System sollen Angestellt nun Videos konsumieren können. Motiviert durch die Tokens kommt es nicht mehr zu Gewissenskonflikten Wissen zu dokumentieren und konsumieren, da man es genau über Pay-per-Time bilanzieren kann. Das Unternehmen profitiert von der genauen Bilanzierung und baut eine nutzbare sichere Wissensbasis auf. Die Daten sind über die Kopplung eines klassischen DRM-Systems und Tokens geschützt, sicher und proprietär an das Unternehmen gekoppelt. Das sichert darüber hinaus die Wiederverwendbarkeit im Unternehmen und domänenspezifische Daten können nicht über das System hinaus genutzt werden.

4 Umsetzung und prototypische Implementierung

Die Umsetzung des prototypischen Systems erfolgt für die Firma adorsys. Daher wird von der in adorsys etablierten Software und Hardware ausgegangen. Jeder Angestellte besitzt einen von der Firma finanzierten Geschäftsrechner, im Normalfall ein MacBook, und ein Smartphone mit Vertrag. Daher wird in der folgenden Implementierung von Apple-Systemen ausgegangen. Das bedeutet, dass die Clients werden als Mac-App und iOS-App entwickelt. Über das „Proof-of-Concept" hinaus wird von einer plattformunabhängigen Nutzung über verschiedene Clients ausgegangen. Außerdem wird davon ausgegangen, dass das Kommunikationstool Slack und Confluence parallel zu dem System existieren und genutzt werden können. Aufgrund von zeitlicher Beschränkung dieser Arbeit wird bei Blockchain mit weltweit etablierten Standards gearbeitet und das System zunächst als „Proof-of-Concept" ausgelegt. Die Dimensionierung der Plattform kann daher reduziert werden. Deshalb sollte das System trotzdem skalierbar und für die entsprechenden Einsatzzwecke erweiterbar sein. Wie in der Konzeption erwähnt gibt es Plattformen zum Schutz von Videos, die allerdings nicht verwendet werden können. Diese Prozesse erfordern Authentifizierungen über das Unternehmen und gehen weit über die Registrierung als App-Entwickler für einen App Store hinaus. Daher wird statt zum Beispiel Apples Fairplay ein einfacher, klassischer REST-Server verwendet, um das DRM zu simulieren. Des Weiteren wird davon ausgegangen, dass bekannte Verschlüsselungsverfahren, wie symmetrische, asymmetrische Verschlüsselung und HTTPS-Verbindungen sicher und nicht angreifbar sind. In der Regel ist aber, ausgehend von unbeschränkten Ressourcen, jedes System angreifbar.

4.1 Blockchain und Token-System

4.1.1 Ethereum

Das Micropayment-System wird auf eine Blockchain durch Tokens aufgesetzt. Hier kommt die etablierte Kryptowährung Ethereum zum Einsatz. Ethereum ist aktuell das

meist genutzte öffentliche Blockchain-Netzwerk mit der größten Community, welches die Entwicklung von individuellen Smart Contracts erlaubt. Der dazu auf GitHub veröffentlichte und etablierte Standard ERC-20 ist ein weltweit weitverbreiteter und viel genutzter Token. (vgl. [ETHEREUM20]) Darüber hinaus gibt es einige für diese Arbeit relevanten Weiterentwicklungen dieser Standards. In der Implementierung wird ERC-20 verwendet, da es sich um eine gute Grundlage für eine späteren Umstieg auf eine ausgearbeitete, eigenständige Weiterentwicklung handelt.

„Ethereum ist eine globale Open-Source-Plattform für dezentralisierte Anwendungen" [ETHEREUM20]. Das Netzwerk besitzt wie andere Blockchains auch eine native Kryptowährung die „Ether" genannt wird. Darüber hinaus ist es programmierbar und erlaubt damit unter anderem die Entwicklung von Smart Contracts, Finanzanwendungen und Kryptowährungs-Accounts. Außerdem wird Ethereum von einer vielfältigen globalen Gemeinschaft betrieben, instand gehalten und basiert nicht auf einer zentralisierten Organisation. Es steht für Privatpersonen als auch als Enterprise Version für Unternehmen zur Verfügung. Für die Enterprise Nutzung wirbt das Netzwerk mit folgenden Versprechungen für Unternehmen:

- Neue Geschäftsmodelle und Wertschöpfungsmöglichkeiten
- Reduzierte Kosten für Vertrauen und Koordination zwischen Geschäftspartnern
- Verbesserte Rechenschaftspflicht im Geschäftsnetzwerk und operative Effizienz
- Unternehmen wettbewerbsfähig und zukunftssicher machen
- Kompatibilität mit öffentlichen Netzen oder berechtigungsbeschränkten, privaten Netzwerken

Diese und viele weitere Gründe machen den Nutzen von Ethereum als Enterprise Version für Unternehmen interessant. Zu den Enterprise Funktionen zählen verschiedenste Ethereum-Erweiterungen in den Bereichen Berechtigungen, Privatsphäre, Sicherheit und auch Entwicklerwerkzeuge. Des Weiteren gibt es bereits Enterprise fokussierte Dienste, Protokolle und Infrastrukturen, die aber hauptsächlich für laufende Systeme und weniger als Entwicklungsbasis geeignet sind. (vgl. [ETHEREUM20])

Zu den Entwicklerwerkzeugen zählt die „Truffle Suite”. Diese vereinfacht die Entwicklung als Testumgebungen für Ethereum und darauf aufbauende Smart Contracts. Das Testnetzwerk Ganache stellt eine konfigurierbare Mock in Form eines Emulators einer Ethereum Blockchain zur Verfügung. In Abbildung 4.1 sind die Zusammenhänge um das Testnetzwerk herum beschrieben. Diese ist bestückt mit verschiedenen Test-Accounts die über Public-Key-Kryptographie gesichert sind. Durch einen RPC-Server, eine bestimmte Serverart eines Client-Server-Modells, kann man diese Mock für das private Netzwerk freigeben.

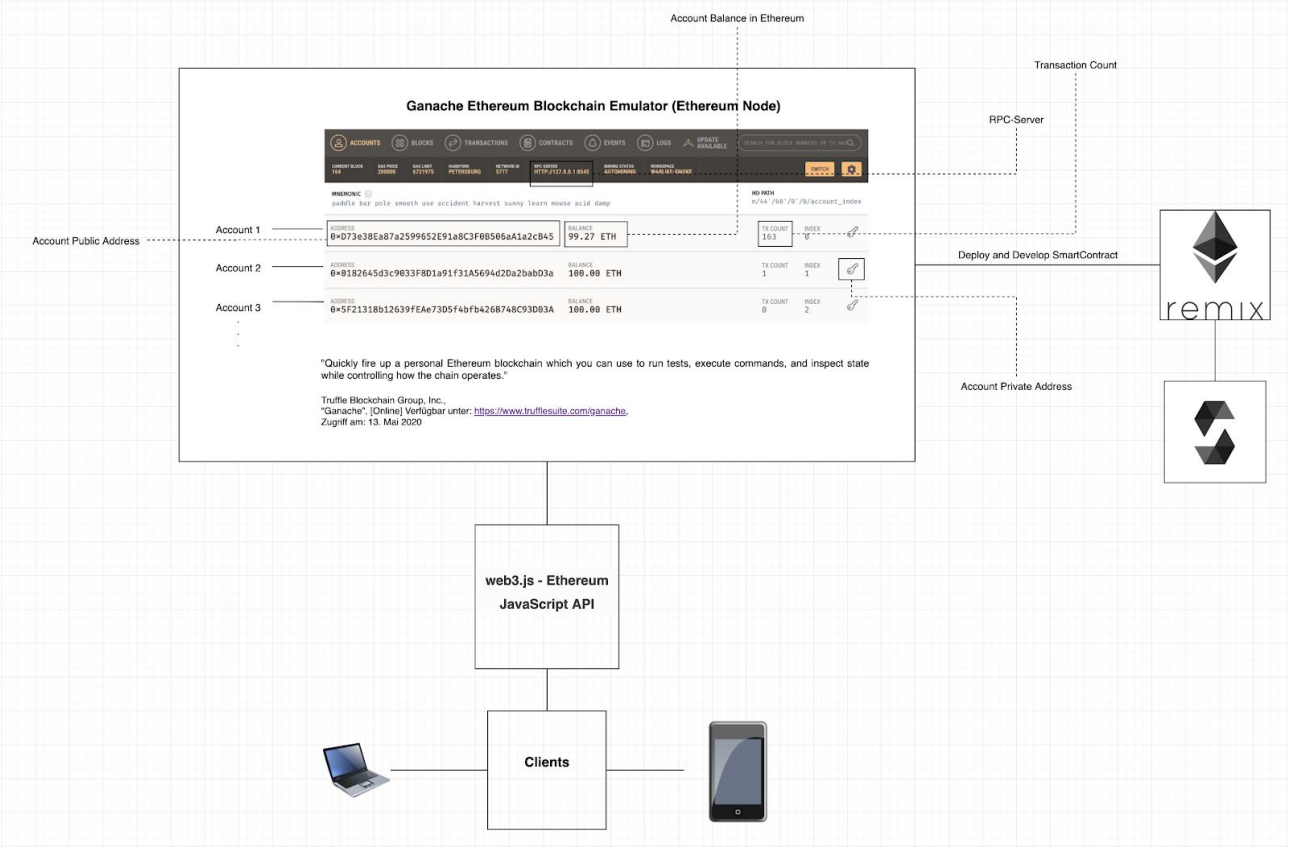

Abbildung 4.1 Entwicklung Smart Contract über Ganache

Um im Netzwerk auf diese Test-Ethereum-Blockchain zuzugreifen kann man die JavaScript-API „web3.js” verwenden. Diese Kollektion verschiedener Libraries erlaubt über HTTP oder IPC die Interaktion mit einer Ethereum-Node im Netzwerk. Das ist nötig um einen Smart Contract auf der Blockchain deployen zu können, um dann über die verschiedenen Clients und Server damit interagieren zu können.

Der Smart Contract kann in der webbasierten Entwicklungsumgebung „Remix" entwickelt und auf der Blockchain deployed werden. Durch den HTTP-Web3-Provider kommuniziert dieser über den RPC-Server mit der in Ganache veröffentlichten Ethereum-Blockchain. In Abbildung 4.1 ist dieser über die URL `HTTP://127.0.0.1:8545` erreichbar. Ein Smart Contract kann durch die spezifische Programmiersprache Solidity geschrieben werden. Ist ein Smart Contract entwickelt, kann dieser mit den gewünschten Parametern durch verschieden Remix-Plugins auf der Blockchain veröffentlicht, genutzt und soweit im Code erlaubt konfiguriert werden. Für diesen Vorgang ist eine Verbindung zu einem Ethereum-Account mit Ether nötig. Denn die Veröffentlichung auf der Blockchain kostet Gebühren, um die Netzwerk-Miner zu bezahlen. Durch dieses Mining-System ist abgesichert, dass das System immer Änderungen annehmen kann. Diese Gebühren sind Nutzungsgebühren und tragen das System, da die asynchronen Veränderungen der Blockchain dauerhaft aktualisierbar sein müssen. Das ist die Rolle der Miner, die sicherstellen das genügend Ressourcen vorhanden sind, um diese Änderungen auf die Blockchain zu bringen. Dieser Nutzungspreis ist abhängig von Angebot, Nachfrage und Umfang des Smart Contracts. Erneut fallen Transaktionsgebühren an, wenn der Aufruf einer Funktion im Smart Contract den Zustand der Blockchain ändert. (vgl. [FRANKENFIELD19]) Darunter fällt auch das Überweisen von Tokens auf unterschiedliche Accounts. Über die Chrome-Erweiterung „MetaMask" können die eigenen Accounts mit einem angenehmen User-Interface verwaltet werden. Hier können Ether und Tokens auch eingesehen werden. Durch Ganache, Remix, Solidity und web3.js sind nun alle Grundlagen gesetzt um auf einen voll funktionsfähigen Ethereum-Mock arbeiten zu können.

4.1.2 Entwicklung ERC-20 Token

Um einen Token zu entwickeln, kann man entweder diesen völlig frei im Rahmen der Blockchain gegebenen Möglichkeiten entwerfen oder man orientiert sich an üblichen Standards wie ERC-20. ERC-20 ist ein fungibler Token von dem es auch schon verschiedenste Abwandlungen gibt. Dieser Token basiert auf objektorientierten Standards.

Der Smart Contract ist in Solidity eine Klasse der Art `contract` und kann je nach Ausführung Funktionen erben. Danach folgen beschreibende Attribute mit der Zugriffsrechten und gegebenenfalls Default-Werten. Bei ERC-20 sind das meist die Attribute `name`, `symbol`, `decimals`, `totalSupply`, `balance` und `allowances`.

`name`	Der Name der Token-Währung.
`symbol`	Das Symbol der Token-Währung.
`decimals`	Die Anzahl der Dezimalstellen der Token-Währung.
`totalSupply`	Die Gesamtanzahl der existierenden Tokens.
`balance`	Die Anzahl der Tokens die nicht im Umlauf sind.
`allowances`	Ein Array von jeweils einer Account-Adressen gemappt auf ein Array einer anderen Account-Adresse mit einem `uint256`. Hier gibt die erste Account-Adresse die Verfügungsgewalt über eine bestimmte Anzahl von Tokens an die Account-Adresse im zweiten Array ab.

Außerdem ist der Stand von `totalSupply`, `balanceOf` und `allowance` über Getter-Methoden abrufbar. Sind die Attribute gesetzt, folgt der Konstruktor, der beim Deployen des Smart Contracts aufgerufen wird. Dieser legt über einen Parameter die Gesamtzahl der existierenden Tokens fest. Darüber hinaus werden die Anzahl der Tokens, die nicht im Umlauf sind, zunächst auf die Gesamtzahl der existierenden Tokens gesetzt. Das ist je nach Implementierung abwandelbar.

Es folgen drei verschiedene Funktionen, um Transaktionen auszuführen. Die erste Funktion `transfer` überweist vom Account des Ausführenden eine über die Parameter festgelegte Anzahl von Tokens an ein anderen bestimmten Account. Nach dem Aufruf der Funktion überprüft der Smart Contract zunächst, ob auf dem Account genügend Tokens zur Verfügung sind, die überwiesen werden können. Ist dies der Fall, zieht der Smart Contract dem Account des Sendenden die Tokens ab und addiert die Anzahl der Tokens auf den Account des Empfängers. Danach werden beide benachrichtigt, dass die Transaktion durchgeführt wurde. Über die nächste Funktion `allowance` kann der Ausführende die Verfügungsgewalt über eine bestimmte Anzahl von Tokens auf seinem Account an ein anderen Account abgeben. Auch hier werden nach erfolgreicher Durchführung alle betreffenden Parteien benachrichtigt. Über die letzte Funktion

`transferFrom` kann ein Dritter, dem die Verfügungsgewalt über Tokens eines anderen Accounts gegeben wurde, von diesem Account einem anderen eine bestimmte Anzahl von Tokens überweisen. Möchte nun der Dritte, der über die bestimmte Anzahl von Tokens verfügen kann, eine solche Transaktion über diese Funktion starten, trägt er die Transaktionskosten und die entsprechenden Parteien werden benachrichtigt. (vgl. [CHITTODA19])

Das Sequenzdiagramm in Abbildung 4.2 beschreibt den Ablauf der Methoden im Smart Contract. Die Accounts und der Smart Contract befinden sich auf der Blockchain. Der Smart Contract kommuniziert über die Accounts der Nutzer. Die Accounts wiederum sind über die Clients mit den Nutzern verbunden. Für die Überschaubarkeit wurde das Überweisen von Ethereum an die Miner aus diesem Diagramm weggelassen. Normalerweise gibt der Nutzer einen bestimmten Ethereum Betrag frei, der genutzt wird, um die Änderungen über die Miner auf die Blockchain zu schreiben. Außerdem wurden in Abbildung 4.2 alle Bedingungen mit `true` behandelt.

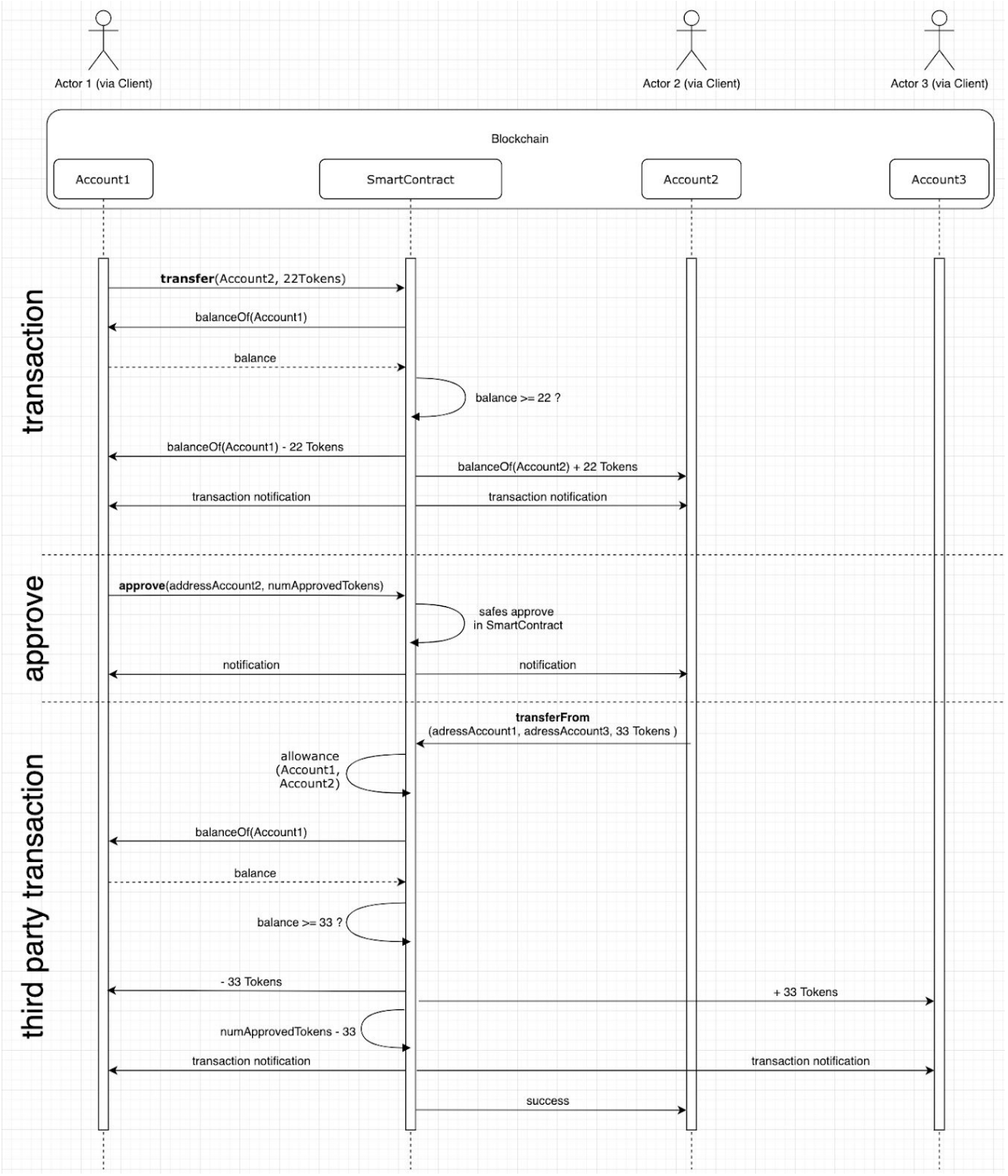

Abbildung 4.2 Sequenzdiagramm Transaktionen ERC-20

Diese Smart Contract Implementierung ist ausbaubar und nicht determiniert. Folglich kann man beliebig Methoden hinzufügen. Es ist zu empfehlen die Implementierung so schlank wie möglich zu halten und Schleifen zu vermeiden, da hoher Rechenaufwand zu hohen Transaktionsgebühren führen kann.

Ist nun der Smart Contract für den individuellen Einsatz fertig entwickelt und getestet, kann dieser deployed werden. Sobald der Smart Contract auf der Blockchain deployed ist, ist es nicht mehr möglich diesen zu verändern. Der Smart Contract Deployer kann nun auf die Accounts seiner Wahl die Tokens über den Aufruf von `transfer` verteilen. Im Fall dieser Infrastruktur kann den Tribe-Accounts jetzt eine bestimmte Anzahl von Tokens nach ermessen des Unternehmens überwiesen werden.

Um Methoden auf dem Smart Contract auszuführen, wird in den Clients und dem Server die Adresse des Smart Contracts benötigt. Über die web3 API und den richtigen RPC-Server kann auf die Blockchain und damit auf den Smart Contract zugegriffen werden. Um mit den Methoden arbeiten zu können, brauchen die drei Anwender eine Datei im JSON-Format ein sogenanntes `abi.json`. Dieses JSON-Datei enthält den Smart Contract und ermöglicht so die Arbeit mit dessen Methoden. Solange man Methoden verwendet, die nichts auf der Blockchain verändern, kann man diese einfach aufrufen. Möchte man etwas auf der Blockchain verändern beispielsweise Tokens überweisen, muss der Anwender sich noch mit seinem Account authentifizieren, um die Gebühren für die Nutzung mit Ether zu bezahlen. Hierzu wird die private Adresse des Accounts benötigt. Nun können, soweit genügend Ether vorhanden, alle Methoden auf dem Smart Contract aufgerufen werden.

4.2 Clients

4.2.1 Verschlüsselung

Eine Anforderung an das System ist, die Videos auf Grund der Dateigröße im Client zu verschlüsseln, um Server-Ressourcen zu sparen. Da die Dateien über einen Dritten, das DRM, verteilt werden, reicht hier eine symmetrische Verschlüsselung und eine Übertragung des Schlüssels. Die Videos in dieser Arbeit werden daher zunächst symmetrisch verschlüsselt. Da beide Arten von Clients auf Rechnersystemen der Firma

Apple basieren, einer Mac-App und einer iOS-App, ergibt es Sinn eine Verschlüsselungsvariante der von Apple vorgeschlagenen Programmiersprache Swift zu verwenden. Daher kommt für diese Umsetzung das von Apple 2019 vorgestellte CryptoKit zum Einsatz. CryptoKit wird von Apple als sichere und effiziente Art eingesetzt, um kryptographische Operationen durchzuführen. Darunter fällt das Vergleichen von sicheren Digests, Public-Key-Kryptographie, symmetrische Schlüssel und vieles mehr. (vgl. [APPLEDEVELOPER20])

Um Cryptokit zu verwenden muss man zunächst das Framework im benötigten File importieren und kann dann auf alle benötigten Methoden zugreifen.

```
import CryptoKit
```

Um nun eine Datei mit einem symmetrischen Schlüssel zu verschlüsseln, muss dieser zunächst erstellt werden. Als Parameter kann man die größe des Schlüssels festlegen. Die Library sieht hier eine Enum aus Gründen der Type-Safety vor der `256bits` entspricht.

```
let key = SymmetricKey(size: .bits256)
```

Danach benötigt man den Pfad der zu verschlüsselnden Datei, um daraus eine `Data` Instanz zu erstellen, welche auf Byte-Ebene verschlüsselt werden kann.

```
let videoPath = Bundle.main.path(forResource: "Video", ofType: "mp4")!
let video = FileManager.default.contents(atPath: videoPath)!
```

Nun kann man die Instanz durch einem vordefinierten Algorithmus mit dem symmetrischen Schlüssel (`key`) verschlüsseln. Apple stellt hier verschiedenste etablierte Verschlüsselungsalgorithmen zur Verfügung, die je nach Anwendungsfall nutzen kann. Darunter fallen beispielsweise AES, Curve25519 und ChaChaPoly. ChaChaPoly wird von Apple, wenn es keinen speziellen Anwendungsfall gibt, als Standard empfohlen wird.

```
let encryptedContent = try! ChaChaPoly.seal(video, using: key).combined
```

Die `encryptedContent` Instanz ist sicher verschlüsselt und kann verschickt werden. Mit dem richtigen Entschlüsselungsalgorithmus kann nun eine sogenannte `sealedBox` erstellt werden und mit dem gleichen Schlüssel entschlüsselt werden.

```
let sealedBox = try! ChaChaPoly.SealedBox(combined: encryptedContent)
let decryptedThemeSong = try! ChaChaPoly.open(sealedBox, using: key)
```

Theoretisch könnte auf ähnlichen Weg eine asymmetrische Verschlüsselung durchgeführt werden. Diese kann verwendet werden, um zum Beispiel für einen Schlüsselaustausch zwischen Client und Server sicher zu kommunizieren. Es wird in dieser Arbeit bei Kommunikation zwischen den Server und Clients von HTTPS-Verbindungen ausgegangen, die eine asymmetrische Verschlüsselung der Requests überflüssig machen würden. Für weitere Sicherheitsschritte können Daten über HMAC authentifiziert werden und über Public-Key-Kryptographie gekennzeichnet werden. Dies wird aber für das Konzept nicht benötigt.

4.2.2 Produzent (Producer)-Client

Nachdem ein Nutzer ein Video produziert hat, kann er dieses auf sein MacBook laden. Die Mac-App wird in Apples Entwicklungsumgebung Xcode in Swift entwickelt. Das Benutzerinterface ist mit der deklarativen Syntax SwiftUI geschrieben und findet sich in Abbildung 4.3 und 4.4 wieder. Das Interface ist klassisch in einem Programmfenster mit zwei zentralen Button zum Auswählen eines Videos und zum Upload dessen gehalten. Darüber ist ein Label in dem, sofern verfügbar, der Pfad der Datei angezeigt wird. Darüber ist ein Vorschaubild des Videos aus dessen Metadaten. Unter den Buttons wird während dem Upload-Prozess ein Fortschrittsbalken angezeigt werden. Über ein Zahnradsymbol in der oberen rechten Ecke können Einstellungen modifiziert werden. Die Mac-App wird nur als Upload-Client fungieren, die iOS-App nur als Download-Client. In einer über eine Konzeptarbeit hinausgehende reale Umsetzung können natürlich beide Clients alle Prozesse implementieren.

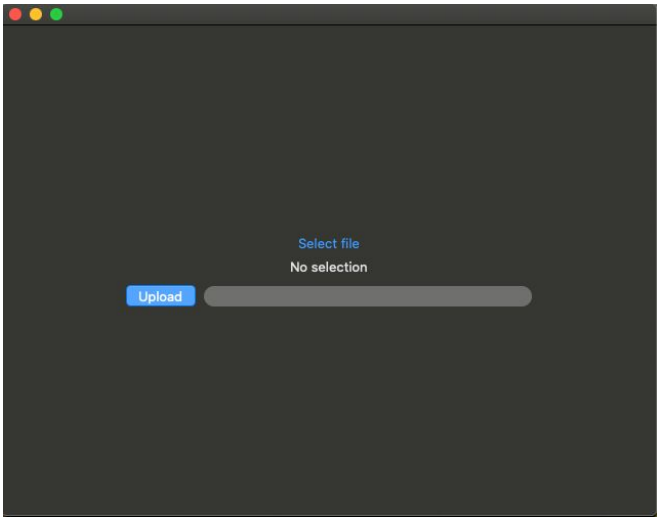

Abbildung 4.3 Bildschirmfoto Mac-App ohne Auswahl

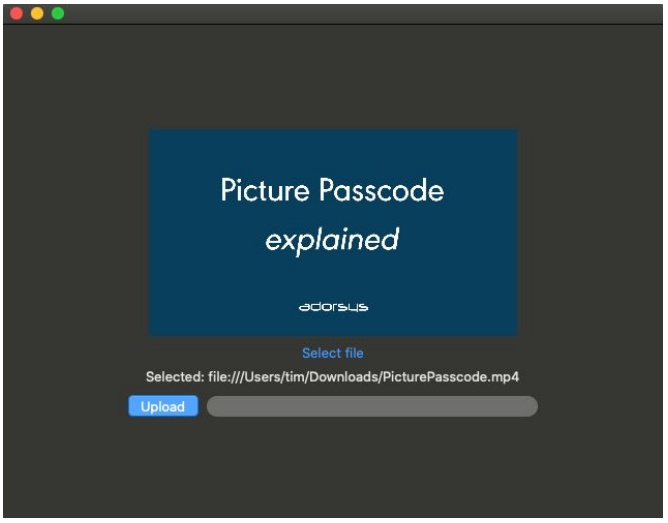

Abbildung 4.4 Bildschirmfoto Mac-App mit Beispiel Auswahl

Der Nutzer kann durch betätigen des „Select file”-Buttons ein Video, welches er hochladen möchte, durch ein Finder-Panel im Mac-File-System auswählen. Dieser Prozess ist in Abbildung 4.5 in einem Aktivitätsdiagramm veranschaulicht und im folgenden näher erläutert. Danach kann er sich über ein Vorschaufenster vergewissern, ob er dieses Video hochladen möchte. Sobald er sich entschieden hat, kann er den Upload-Button betätigen. In dieser Implementierung ist ein Beispiel Account vorprogrammiert in einer realen Umsetzung könnte ein PopUp erscheinen mit dem er die Adresse des Accounts übergibt auf dem die Tokens später überwiesen werden sollen. In dieser Implementierung ist das ein normales Textfeld. In einer

Real-Implementierung könnte hier zur Vereinfachung ein QR-Code-Scan starten. Sofern authentifiziert wird der symmetrische Verschlüsselungsprozess mit Hilfe von CryptoKit durchgeführt. Ist dieser asynchrone Prozess abgeschlossen, wird der Upload der verschlüsselten Datei gestartet. Hier kommt Alamofire zum Einsatz, ein Framework zum einfachen Networking unter Swift. Über den Aufruf von `Alamofire.upload()` wird die Datei an einen Endpoint hochgeladen. Während des Uploads wird über Alamofire ein Fortschrittsbalken aktualisiert. Nach Erfolg erhält der Client ein Status-Code 200, der den Erfolg bestätigt. Der Nutzer sieht eine Bestätigung durch das Vollenden des Fortschrittsbalkens. Mit einem separaten Request wird die öffentliche Account-Adresse zusammen mit dem symmetrischen Schlüssel übergeben. Der symmetrische Schlüssel wird zuvor noch in eine textbasierte Form umgewandelt, damit dieser auch Programmiersprachen-agnostisch von unterschiedlichen Systemen verarbeitet werden kann. Damit ist der Upload-Prozess abgeschlossen.

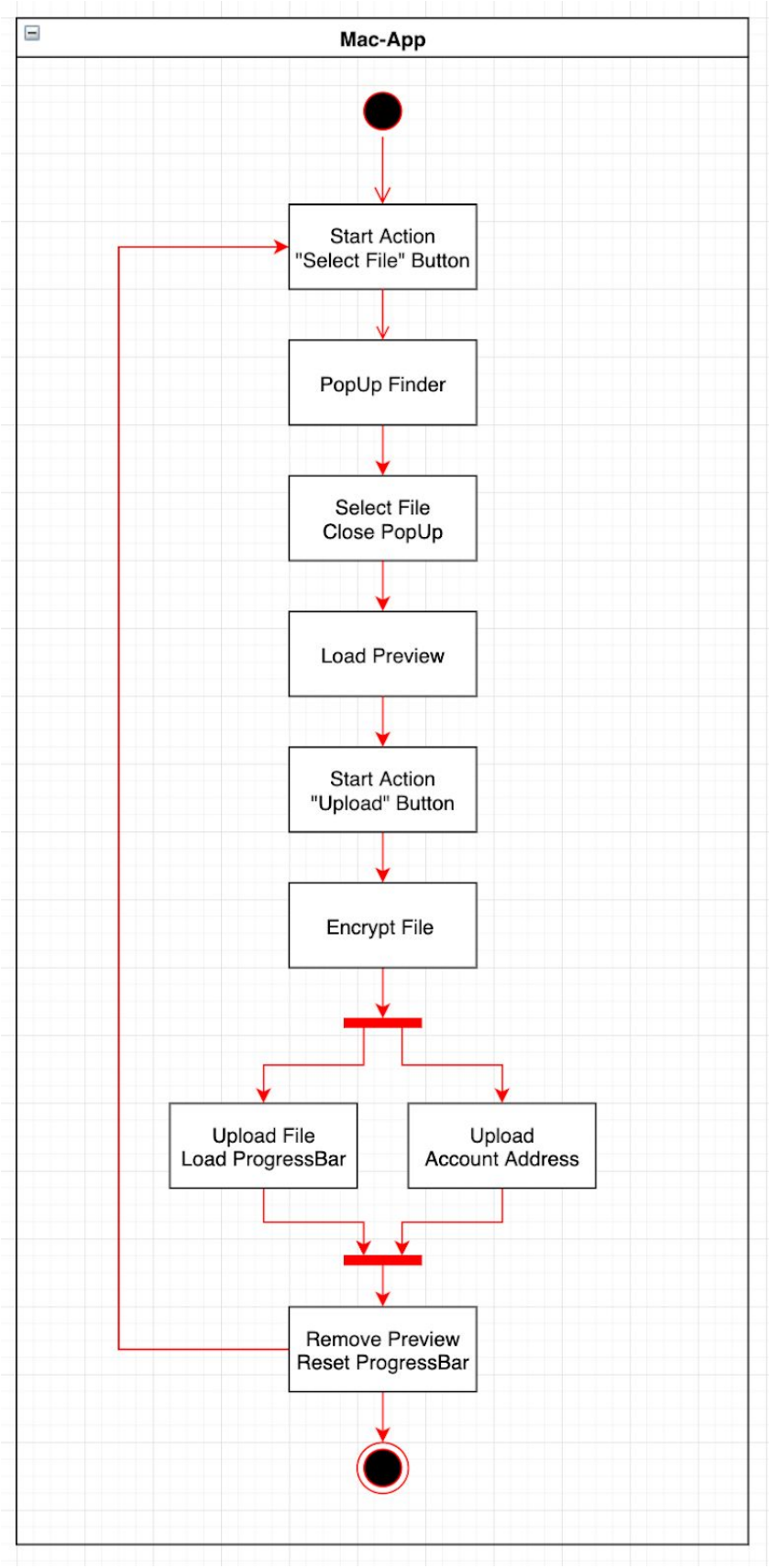

Abbildung 4.5 Aktivitätsdiagramm Mac-App

4.2.3 Konsument (Consumer)-Client

Ein Konsument möchte ein Video konsumieren. Dies soll er von jedem beliebigen Ort im Rahmen der Netzwerk Öffnung des Systems erledigen können. Aus diesem Grund kommt daher eine iOS-App zum Einsatz, da jeder Angestellte sein Smartphone immer bei sich trägt. Das Interface der App ist nach den Human-Interface-Guidelines für iOS-Apps gestaltet und nicht in SwiftUI, sondern klassisch mit UIKit durch Storyboards und MVC entwickelt. Auf dem ersten Storyboard ist eine Anmeldung durchzuführen. Hier kommt ein klassischer Login zum Einsatz, der durch Labels, Inputfeldern und Button gestaltet ist. Auch hier ist eine Authentifizierung durch den öffentlichen Schlüssel der Account-Adresse durchzuführen. Auch das kann durch die Einbindung von QR-Code-Scans verbessert werden.

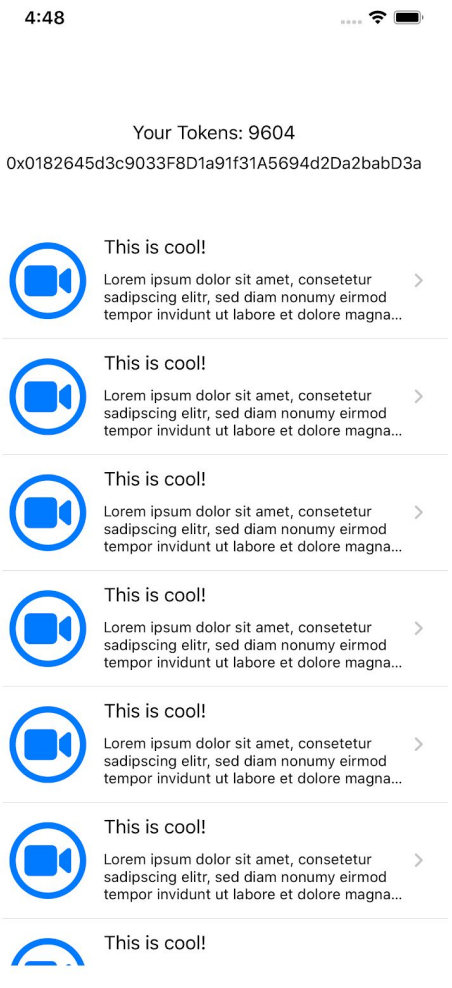

Abbildung 4.6 Bildschirmfoto iOS-App mit Beispielinhalt

Der zweite Screen in Abbildung 4.6 ist primär durch eine Liste gestaltet. In dieser werden die Videos des Servers aufgelistet und mit deren Metadaten aufgeführt. Hierzu muss die App mit dem Server kommunizieren und die Daten über einfache Requests abrufen. Das kann allerdings ohne Authentifizierung erfolgen, da keine für die Firma kritischen Inhalte in diesem Schritt zu sehen sein sollten. Über die Liste kann der Nutzer sich für ein Video entscheiden und dieses auswählen.

Hat er sich für ein Video entschieden, kann er dieses auswählen und bezahlt im Hintergund mit den Tokens. Dieser Prozess mit der Kommunikation mit dem DRM ist ausführlich im Aktivitätsdiagramm in Abbildung 4.7 beschrieben. Der Stream des Videos wird gestartet. Da Apple-Fairplay nicht zur Entwicklung zur Verfügung stand, wird nun mit einem klassischen Download der entsprechenden Datei gearbeitet. Ausgehend davon, dass der Download der entsprechenden Datei über einen Alamofire Download-Request erfolgt und abgeschlossen ist, kann nun die Datei entschlüsselt werden. Mit dem zugehörigen Schlüssel, empfangen über einen separaten Request, wird über CryptoKit das Datenpaket entschlüsselt und das Video kann angesehen werden.

4.3 Digital-Rights-Management

4.3.1 REST

Für den Prototypen kommt ein klassischer REST-Server zum Einsatz. Da bis jetzt der gesamte Prototyp auf Swift basiert, macht es Sinn Swift-Server zu verwenden. Leider wurde die Weiterentwicklung von Swift-Server eingestellt und ist damit für den Use-Case als zukünftiges System schlecht nutzbar. Um mit einen vielfältig konfigurierbaren Server zu arbeiten, wird im Prototyp mit `node-express` gearbeitet. Dadurch kann ein einfacher REST-Server mit entsprechenden Endpoints aufgesetzt werden. Hier kommen folgende relative URLs zum Einsatz über die die Clients mit dem Server über Requests kommunizieren.

- `get  -> /`
- `get  -> /download`

- post -> /symmetricKey
- post -> /payForContentReceiveSymmetricKey
- post -> /upload

Möchte der Producer-Client nun eine Datei hochladen, ruft dieser wie erwähnt über das Alamofire Framework den `/upload`-Endpoint durch einen POST-Request auf und der Server empfängt die Datei, wie im Aktivitätsdiagramm in Abbildung 4.7 verdeutlicht. Die Datei wird für die Einfachheit des Prototypen in das Laufzeit-Verzeichnis des Servers geschrieben und beispielhaft wird der Pfad der Datei in eine Variable `encryptedFile` geschrieben, um damit weiterarbeiten zu können. Der Server antwortet nach erfolgreichem Upload mit der Nachricht `success`. Mit der Nutzung und für das Heraussuchen eines Videos, muss der Client über die entsprechenden Metadaten der Videos verfügen. Diese erhält der Client über einen normalen GET-Request auf den Server. Über den `/download`-Endpoint kann die verschlüsselte Datei über einen GET-Request heruntergeladen werden.

Über den Endpoint `/payForContentReceiveSymmetricKey` kann der Client nun den zugehörigen Schlüssel zur Entschlüsselung herunterladen. Hierzu muss der Client sich wie in Kapitel 4.1.2 beschrieben Methoden auf dem Smart Contract aufrufen können. Bevor der symmetrische Schlüssel gesendet wird, wird überprüft ob der Consumer-Client Tokens besitzt. Ist dies der Fall wird überprüft ob dem Server genügend Tokens zur freien Verfügung gestellt wurden. Für den Prototypen wurden alle Tokens von anderen Accounts dem Organisations-Account zur Verwaltung freigegeben. Darauffolgend wird die Überweisung über `transferFrom(ConsumerAccountAddress, ProducerAccountAddress, TokenNumber)` getätigt und der Nutzer erhält den symmetrischen Schlüssel. Diese Transaktion ist eine Ethereum Transaktion in welcher der Smart Contract aufgerufen wird. Für die Überweisung ist die Authentifizierung mit private Key des Organisations-Accounts nötig, da die Überweisung auf der Blockchain vermerkt wird, damit sich etwas auf dieser ändert und dieser Prozess Ether kostet. Bei einer realen Umsetzung würde diese Schnittstelle für eine Abrechnung über Pay-per-Use oder Pay-per-Time fungieren.

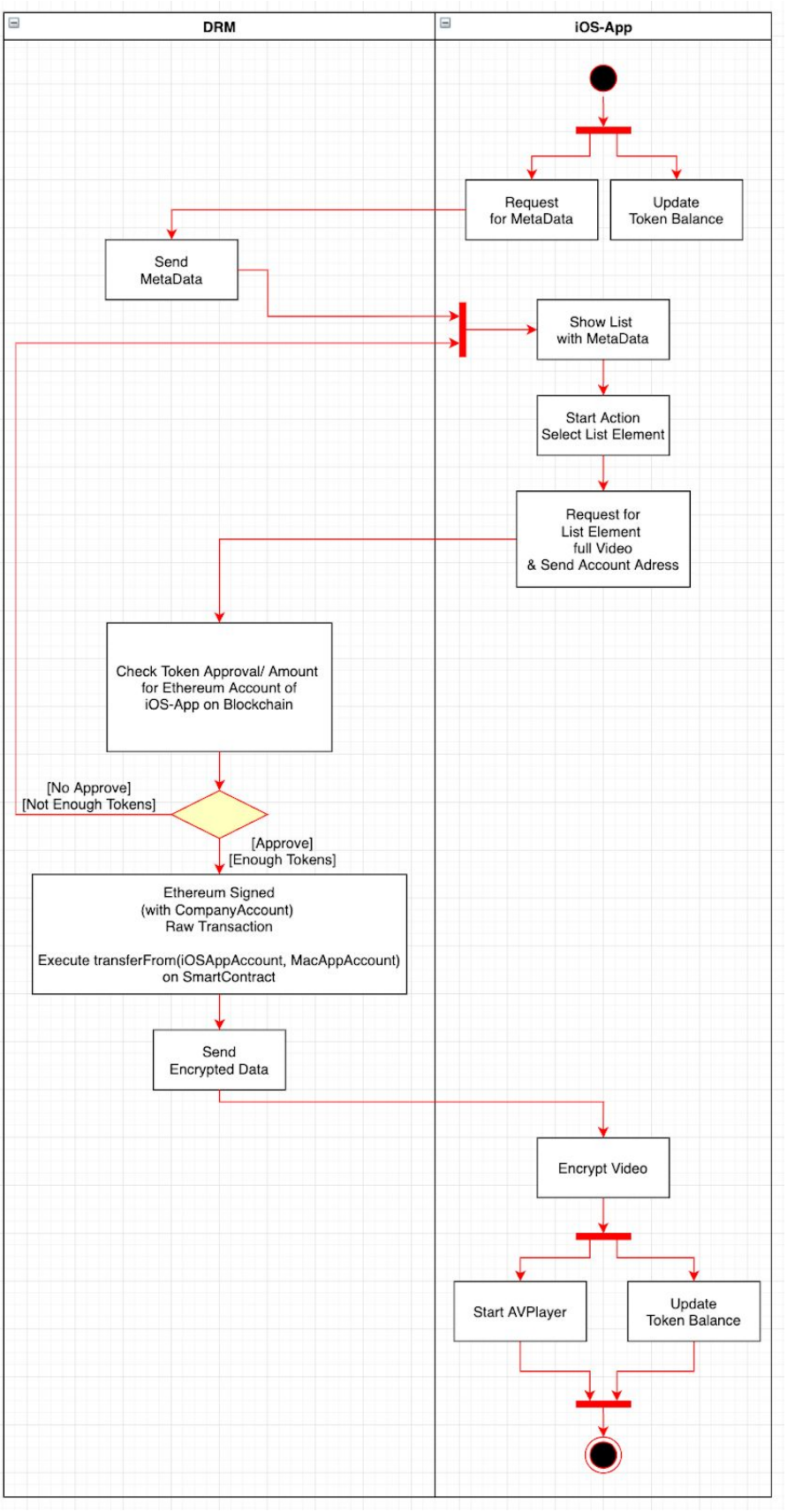

Abbildung 4.7 Aktivitätsdiagramm DRM und iOS-App

4.3.2 Verschlüsselung

Die Verbindungen zwischen den verschiedenen Endgeräten und dem Server erfolgt über HTTPS. Hierzu wird ein Zertifikat und ein zugehöriger Schlüssel benötigt die aus dem Dateisystem gelesen werden. Mit der Konfiguration `options` kann der express Server `app` als HTTPS-Server auf einem beliebigen Port gestartet werden. Es macht Sinn ein solches SSL-Zertifikat zu kaufen und nicht selber zu erstellen, damit herkömmliche Browser die URL nicht blocken. Mit den Node-Libraries `elliptic` und `eccrypto` sind auch eine symmetrische und asymmetrische Verschlüsselung leicht umzusetzen. Theoretisch könnten hierdurch wichtige Informationen noch einmal verschlüsselt werden bevor diese in einer Datenbank abgelegt werden. Allerdings ist hier die Frage ob ein potentieller Angreifer, der die Daten erreicht auch den Verschlüsselungsprozess auf dem Server erreicht und damit die Daten entschlüsseln könnte. Über das „Proof-of-Concept" hinaus sollte in diesem Schritt auf ein gängiges DRM zurückgegriffen werden. Das Framework Apple-Fairplay behandelt diese Probleme und wird von Apple mit den aktuellsten Sicherheitsvorkehrungen gesichert.

4.3.3 Datenbanken

Die Dateien, die im Prototypen im Dateisystem der Laufzeitumgebung des Servers gehalten werden, können natürlich in einer realen Umsetzung nicht dort gespeichert werden. Zum einen wären mit einem Serverabsturz alle Daten verloren, zum anderen würde das mit steigender Datenanzahl theoretisch unendlich Ressourcen und Rechenleistung benötigen. Daher ist es notwendig auf Datenbanken und File-Server zurückzugreifen. Bei Cloud-Lösungen muss zunächst der rechtliche Rahmen und die Sicherheit für den Nutzungszweck überprüft werden. Auf diese Systeme würde über den Server zugegriffen und die Dateien verwaltet werden.

Für Videos ist ein theoretisch unendlich skalierbarer File-Server nötig, der darauf ausgelegt ist große Dateien zu speichern. Ein kommerzielles Beispiel hierfür wäre Amazon S3, welches ein solches File-Hosting System ist und nach Verbrauch abrechnet. Für symmetrische Schlüssel und Metadaten wären klassische, entsprechend gesicherte Datenbanken nötig. Ein Beispiel wären SQL-Datenbanken.

Darüber hinaus wäre bei der Verwendung von asymmetrischen Schlüsseln noch ein Key-Server nötig, der die öffentlichen Schlüssel zur Verfügung stellt. In diesem hier entwickelten System wird die öffentliche Account-Adressen des Producers mit dem Video hochgeladen. Der Server kann diese entweder mit den Metadaten des Videos verwalten, oder ebenso mit einem Key-Server mit Arrays von zugehörigen Videos und öffentlichen Account-Adressen.

5 Diskussion des Konzeptes

Das Value Proposition Canvas (siehe 3.2) hebt die Vorteile des Systems hervor. Aber bietet das System einen realen Vorteil gegenüber herkömmlichen Systemen für Wissenstransfer? Im folgenden ist eine theoretische Diskussion mit der Auseinandersetzung über das Potential des Systems. Über den Rahmen der Arbeit hinaus sollten noch Usability Tests und verschiedene weitere Tests auf Basis der theoretischen Betrachtung durchgeführt werden, um die Ergebnisse zu validieren.

5.1 Anreizsystem

Einen Mehrwert bietet das von proprietären Token gesteuerte Anreizsystem. Dadurch ist das System selbsttragend und bietet über den Sinn von herkömmlichen Wissenstransfer hinaus den Nutzern noch Anreize. Aus wirtschaftlicher Sicht ist das System ein Schritt zur Integration eines Wissentransfersystems in ein agiles Unternehmen. Darunter fällt, dass das System die Kosten von Wissenstransfer durch die Kombination aus kommerzieller Plattform und eigenem Lehrmaterial transparenter und steuerbar macht. Hierunter fällt die Bilanzierung von Konsum und Erstellung von digitalen Gütern, die den Nutzern helfen diese ordnungsgemäß durch das System zu dokumentieren und gewinnbringend zu nutzen. In diesem Kontext wird aber davon ausgegangen, dass die Nutzer ein solches System auch nutzen wollen und die Vorteile des Systems ausschöpfen. Umso breiter die Nutzer Masse um so einfacher die erfolgreiche Ausschöpfung des Anreizsystems, da nur, wenn auch Nutzer die Videos schauen, sich die Produktion von Videos lohnt. Umso mehr potentielle Nutzer umso wahrscheinlicher, dass es einen Nutzer gibt der ein produziertes Video sehen möchte. Dasselbe gilt für das Spektrum an Videos im System, gibt es schon viele Videos, ist es wahrscheinlicher ein Video zu finden das für einen potentiellen Nutzer interessant ist.

Diese These geht folglich davon aus, dass sobald es ein breites Spektrum an Videos gibt und eine hohe potentielle Nutzerzahl das System besser funktioniert. Dadurch könnte es in einem mittelständischen Unternehmen mit noch nicht vielen Mitarbeitern schwieriger werden ein solches System zum Anlaufen zu bringen.

Ein weiterer Punkt ist der domänenspezifische Inhalt, der meistens für eine große Anzahl der Nutzer im System von Nutzen sein sollte. Aber es gibt Randthemen, die unter Umständen nur wenige Nutzer betreffen. Hier könnte es für den produzierende Nutzer wenig Anreize geben ein Video für ein solches Randthema zu produzieren, da nur wenige ihm dafür Geld bezahlen würden. Entsprechend würde der Preis steigen. Auf der anderen Seite funktioniert die Produktion auf Arbeitszeit und der Nutzer wird dafür bezahlt. Hier sollten Schranken gegeben werden wie viel Prozent der Arbeitszeit für die Produktion von Videos fungieren dürfen. Dies ist aber individuell von Mitarbeiter zu Mitarbeiter anders. Des Weiteren schützen Schranken das System vor Ausnutzung und es sollte eine kontrollierende Stelle geben, die darauf achtet, dass die produzierten Videos nicht sinnlos sind. Diese Regeln sind auch auf den genauen Nutzungskontext abstimmbar.

Durch Einweisungen in alle systemabhängige Strukturen und zusätzliche Anreize sollte zu Beginn die Einstiegshürde für Nutzer gemindert werden. Auch für den Fall, dass ein Nutzer keine Videos produzieren möchte, müssen Regelungen geschaffen werden. Das fordert zu Beginn einen Mehraufwand bis ein solches System zur allgemeinen Adaption in einer Firma kommen kann. Aber sobald das System adaptiert ist, sollte es sich dauerhaft selbst tragen und durch das Anreizsystem funktionieren.

Eine Öffnung des Systems für den offenen Internetverkehr könnte es stark antreiben und für die Organisation eine gewinnbringende Erweiterung sein durch die sie das in der Organisation entwickelte Wissen verkauft. Die Preise nach außen hin müssten entsprechend dem Wert des Inhalts berechnet werden. Hier wäre durch das Internet sofort eine potentiell riesige Nutzerzahl verfügbar, die das System durch das Anreizsystem antreiben würden. Allerdings ist durch das domänenspezifische Wissen schwierig zu kontrollieren was öffentlich einsehbar sein darf und was nicht. Ein Kompromiss wäre das System zu teilen in Themen, die nur firmenintern zu halten sind und in Themen welche öffentlich im Internet verfügbar sein können. Die Firma könnte

sich hier direkt über das Token-System von Endnutzer bezahlen lassen. In diesem Fall müssten Nutzer, die den öffentlichen Content nutzen wollen, zuvor Tokens kaufen. Dennoch Bedarf eine Öffnung des Systems nach außen noch vielen Testdurchläufe bis eine eindeutige Aussage getroffen werden kann.

5.2 Ressourcen

Damit die Infrastruktur bestmöglichen optimiert für den Gewinn aller Nutzer ausgelegt ist, kommen die proprietären Tokens über Micropayments zum Einsatz. Diese sollen nach dem Prinzipien Pay-per-Use oder Pay-per-Time eingeteilt werden. Pay-per-Use hat für den konsumierenden Nutzer den Vorteil, dass man sich die Nutzungsrechte an dem digitalen Produkt kauft und dieses so oft nutzen kann wie der Nutzer möchte. Dafür hat es den Nachteil, wenn man das Produkt nicht öfter nutzt, dass es im Vergleich zum Nutzen über Zeit teurer ist, da berechnet wird das es öfter genutzt wird. Pay-per-Time bietet darüber hinaus den Vorteil, dass auch wenn der Nutzer nur einen Teil des Produkts nutzt dies billiger ist, da dieser dann auch weniger Zeit mit dem Produkt benötigt. Dies sind die wichtigsten wesentlichen Unterschiede zwischen Pay-per-Use und Pay-per-Time in diesem System. Ein Kompromiss könnte die Vorteile beider Zahlarten verbinden. So kann ein Tribe ein Produkt via Pay-per-Use für die Nutzung aller Tribe-Mitglieder für einem bestimmten Zeitabschnitt kaufen, ein Produkt in welchem ein Nutzer nur kurz einen bestimmten Teil nutzen möchte, kann durch Pay-per-Time konsumiert werden. Das macht das System für die Mehrfachnutzung effizient preiswerter.

Durch die Proprietarität kann der Wert eines Tokens durch die Organisation festgelegt werden. Dieser sollte für ein offenes System abhängig vom Content festgelegt werden. Spezielle Inhalte können teurer, einfache Inhalte billiger sein. Dies sollte nach Angebot und Nachfrage hin angepasst werden. Für eine Nutzung innerhalb der Firma sollte für eine möglichst weite Nutzung und Verteilung der Inhalte möglichst kostengünstig zur Verfügung stehen. Für eine optimale Nutzung kosten hier abhängig von der Nutzungszeit alle Inhalte gleich viel.

Mit der Nutzung eines Smart Contract über das öffentliche Ethereum Netzwerk sind die Nutzungsgebühren für die Miner notwendig, um Änderungen auf der Blockchain

oder im Smart Contract zu vermerken. Diese müssen über die Accounts der Tribes oder des Unternehmens bezahlt werden. Angenommen zu Beginn der Nutzung des Systems würden alle Tribes über ihre Accounts dem Unternehmen die Verfügungsgewalt über die Tokens geben, dann könnte das Unternehmen über dessen Account alle Nutzungsgebühren bezahlen (siehe 4.1.2). Pro getätigte Token Überweisung wären das circa 2€ je nach Ethereum Kurs. Das ist für ein Micropayment basiertes System sehr teuer und ineffizient. Dafür würde das Unternehmen sich die Ressourcen sparen eine eigene Blockchain zu betreiben, die nur mit vielen Knotenpunkten sicher wäre, das wäre auch mit sehr hohen Kosten verbunden. Durch eine Öffnung des Systems nach außen könnte das Unternehmen die Gebühren für Nutzer außerhalb der Firma erheben und damit alle Nutzungskosten decken und darüber hinaus mit diesen Nutzern Gewinn zu generieren.

Die Produktion und der Konsum von Videos oder allgemein digitalen Gütern für das System soll für die Mitarbeiter des Unternehmens auf Arbeitszeit gebucht werden können. Das erfolgt mit den Regelungen, die zuvor schon für Wissenstransfer und Selbststudium gegolten haben. Die Betreuung des Systems für Produktion und Instandhaltung der Ressourcen und Anwendungen über die Blockchain hinaus müssen abgedeckt werden, sowie die Ausstattung von Räumen zur Produktion von Videos.

5.3 Technische Betrachtung

Um die technische Sicherheit des Systems zu prüfen werden in der folgenden Aufzählung potentielle Risikopunkte für die Nutzungen des Systems ausgelegt.

- Dauerhafte Systemverfügbarkeit ist gegeben.
- Die Infrastruktur ist theoretisch unendlich skalierbar.
- Wenig Pflegeaufwand durch Blockchain, Apple-Fairplay und Datenbanken über Amazon.
- Allerdings müssen die Clients aktuell gehalten werden.
- Die Quote der Softwarefehler sollte durch viele Tests gemindert werden.
- Durch die proprietären Tokens kann die Qualität und der Nutzen des Systems gemessen werden.

- Keine Behinderung von Altprozessen im Unternehmen, sondern nur zusätzliches Angebot für die Nutzer.
- Nur zwei wichtige Schnittstelle für Clients (Konsum und Produktion), das DRM verwaltet alle anderen Schnittstellen.
- Die technischen Prozesse weisen einen weiten Automatisierungsgrad auf.
- Durch die Blockchain und damit den Tokens im System, werden durch die Bezahlung die Zugriffe auf die Daten dezentral abgesichert.
- Die Integration neuer Anwendungen sollte über die Schnittstellen einfach erfolgen können.

(vgl. [KRANKENHAUSIT20])

Die aufgeführten Punkte zeigen, dass das System eine ausreichend weite technische Dauerhaftigkeit und Sicherheit aufweist. Mit der Verwendung von Apple-Fairplay für den Videostream, durch die Blockchain gesicherte Verwaltung von Tokens und die Herausgabe von digitalen Gütern nur gegen Tokens weist das System eine hohe technische Sicherheit auf. Aber jedes System ist angreifbar und es kann immer Schwachstellen geben in den potentielle Angreifer infrastrukturelle Schwächen ausnutzen und Systeme überlisten.

5.4 Andere Plattformen im Vergleich

Neben dem in dieser Arbeit beschriebenen System gibt es viele verschiedene Systeme für Dokumentation. Anzuführen ist hier, dass in adorsys etablierte System Confluence. Confluence bietet die Erschaffung einer zentralen Informationsquelle in allen Arten von Medien basierend auf Bild und Schrift. YouTube ist eine etablierte Plattform zum Vertrieb von öffentlichen Videos. Auch auf dieser Plattform können Videos privat verwaltet und nur für bestimmte Nutzer über ein Google-Konto freigegeben werden. Hier gibt es die Möglichkeit über Werbung Geld zu verdienen. Außerdem gibt es Systeme wie Wisita, welche die Betreibung einer privaten Video-Plattform ermöglichen. Es stellt sich die Frage warum ein System, wie in dieser Arbeit beschrieben, genutzt werden sollten, wenn es schon etablierte Systeme gibt. Systeme wie Confluence schließen sich mit dem in dieser Arbeit entwickelten System nicht aus, da Confluence vermehrt auf Schrift und Bild setzt und nicht um digitale Güter, wäre eine parallele Nutzung beider System sinnvoll. Das System dieser Arbeit ist ein

System zur Verwaltung von alleinstehenden digitalen Gütern und der Bilanzierbarkeit dieser. Genau dieser Punkt hebt das System von anderen Systemen zum Wissenstransfer ab. Des Weiteren unterscheidet sich das System von anderen Systemen zur Verwaltung von digitalen Gütern durch die Blockchain mit den proprietären Tokens und bietet viele Sicherheitsvorteile. Ausgehend von Videos als digitales Gut unterscheidet sich das System von YouTube, Wisita und anderen Systemen dadurch, dass wichtige domänenspezifische Inhalte auf eigenen Datenbanken gesichert werden können und nicht auf Datenbanken von Drittanbietern, für die es oft in der Dienstleistungsbranche von Kunden vorgelegte Regelungen gibt. Außerdem bietet das neu entwickelte Anreizsystem in der Infrastruktur Vorteile durch die das System das Unternehmen intern weiterbringen kann. Für diese Spezialfälle macht das in dieser Arbeit beschrieben System Sinn. Für digitale Güter oder Videos bei denen der Inhalt und die Bilanzierbarkeit und der Anreizfaktor nicht für das Unternehmen wichtig sind, können auch andere Systeme funktionieren.

6 Zusammenfassung

Das Zusammenwachsen von virtueller und physikalischer Welt macht es notwendig Güter und Services mit digitalen Währungen verknüpfen zu können. Hierbei ist es notwendig, dass ein Vertrag zwischen dem Serviceprovider und einem anonymen, jedoch referenzierbaren, Individuum geschlossen wird. Abhängig vom Kontext, kann es von großer Bedeutung sein, dass die Gebühren, die bei einem solchen System auftreten, sich möglichst in Relation zum wirtschaftlichen Umsatzpotenzial und Geschäftsmodell halten. Am Beispiel einer Infrastruktur für ein firmeninternes domänenspezifisches Wissenstransfersystem in einer Tribe-basierten Organisation wurde diese Herausforderung untersucht und ein Lösungsansatz entworfen.

Damit stellt diese Arbeit ein neues Sicherheitskonzept zur Verarbeitung von Gütern mit Hilfe von Blockchain vor. Ausgehend von einem Value-Proposition-Canvas werden die Werte ermittelt, die festlegen, wie ein solches System funktioniert und wie es Mehrwert neben anderen Systemen schafft. Das System ist entworfen für ein neues bilanzierbares Sicherheitskonzept zur Verwaltung von digitalen Gütern. Auf Basis von an die Blockchain gebundenen proprietären Tokens und Smart Contracts als DRM-Steuerung, kann ein Content-System angelehnt an On-Demand-Prinzipien durch

Micropayments bilanzierbar werden. Die Infrastruktur bestehend aus der Kopplung von klassischen DRM-Funktionalitäten und Funktionalitäten von Smart Contract fähigen Blockchains verbindet die Vorteile der beiden Systeme miteinander. Für die Organisation ist die durch die Bezahlung abgesicherte Bereitstellung der Daten ausschlaggebend.

Der prototypische Entwurf ist serverseitig durch einen JavaScript REST-Server an die Blockchain angebunden über die die Tokens verwaltet werden. An diesen Endpoint greifen clientseitig eine dafür entwickelte Mac-App und eine iOS-App zu. Über diese werden die digitalen Güter verschlüsselt und sicher ausgetauscht. Auf der Ethereum Blockchain wurden über Ganache, web3 und Solidity in Remix, die benötigten Smart Contracts entwickelt.

7 Ausblick

Um das in dieser Arbeit vorgestellte Konzept in der Realität nutzen zu können, muss das DRM-System weiter ausgebaut werden und sollte auf der Basis eines herkömmlichen DRM-Frameworks aufgebaut werden. Dies bietet zusätzliche Sicherheit und Erleichterungen in der Wartung durch die Drittanbieter. Außerdem müssen vor allem Datenbanken und Serverressourcen, die die Skalierbarkeit des Systems beeinträchtigen könnten, für eine optimale Skalierung des Systems weiterentwickelt werden. Abgesehen von der technischen Umsetzung ist die Nutzerfreundlichkeit des Systems noch nicht durch Tests bestätigt. Nutzertest und die damit verbundenen Verbesserungen im User-Interface für eine optimale User-Experience müssen noch durchgeführt werden. Dieser iterative Prozess kann die User-Freundlichkeit verbessern. Des Weiteren muss auch überprüft werden, ob das in dieser Arbeit beschrieben Anreizsystem für den Endnutzer funktioniert. Letztendlich hängt die Nutzung des Systems aber von gut nutzbaren Inhalt ab, der selbst durch die Nutzer produziert und in das System gespeist wird. Nur wenn die Nutzer guten für andere Nutzer notwendigen Inhalt produzieren, sichert dass die Nutzung des System. Eine Herausforderung für das System wird die Einstiegshürde zu überwinden bis genügend Inhalte zur Verfügung stehen, die viele Nutzer des Systems nutzen wollen und können.

8 Fazit

Das in dieser Arbeit entwickelte Konzept funktioniert technisch. Bis zur Adaption eines solchen Systemes für eine bestimmte Nutzungsumgebung sind aber noch weitere Entwicklungsschritte zu bewältigen. Dennoch kann das vorgestellte Konzept funktionieren und trägt einen Teil zur Entwicklung der oft eingeschränkten Benutzerfreundlichkeit von Blockchain-Systemen bei. Das durch die Infrastruktur geschaffene Anreizsystem soll Firmen helfen für Angestellte den Wissenstransfer besser und ressourcenschonender zu gestalten. Für die Firma ist vor allem die Bilanzierbarkeit und die Schaffung einer abgesicherten Wissensbasis interessant. Ob das System wirklich für eine dauerhafte Nutzung ausgelegt ist, kann erst durch viele Tests erwiesen werden.

9 Literaturverzeichnis

[RUMP19] J. Rump, S. Eilers, „Die vierte Dimension der Digitalisierung: Spannungsfelder in der Arbeitswelt von morgen”, Berlin, Springer-Verlag, 2019, S.10

[LINDNER19] D. Lindner, „Was ist das Spotify Modell und wie agil ist es?”, [Online] Verfügbar unter: https://agile-unternehmen.de/was-ist-das-spotify-modell/, Zugriff am: 13. Mai 2020

[WARDT20] R. van der Wardt, „Das Spotify Modell: Agile und Scrum für große Organisationen”, [Online] Verfügbar unter: https://agilescrumgroup.de/spotify-modell/, Zugriff am: 13. Mai 2020

[ADORSYS20] adorsys GmbH & Co. KG, „Was ist adorsys?”, [Online] Verfügbar unter: https://adorsys.de/unternehmen/, Zugriff am: 13. Mai 2020

[NIKODEMUS17] P. Nikodemus, „Lernprozessorientiertes Wissensmanagement und kooperatives Lernen: Konfiguration und Koordination der Prozesse”, Wiesbaden, Springer-Verlag, 2017

[TANDEMPLOY19] Tandemploy GmbH, „Studie zum Wissenstransfer in Unternehmen: Mitarbeitende wollen – werden aber durch Strukturen daran gehindert”, [Online] Verfügbar unter: https://www.tandemploy.com/de/blog/umfrage-wissenstransfer-in-unternehmen/?utm_content=104767294&utm_medium=social&utm_source=facebook&hss_channel=fbp-424355971008796, Zugriff am: 13. Mai 2020

[DOMBRET08] B. Dombret, „Zahlungssysteme im Internet: Marktüberblick und Perspektiven”, o.O., Books on Demand, 2008, S. 14-15

[DANNENBERG04] M. Dannenberg, A. Ulrich, „E-Payment und E-Billing“, Wiesbaden, Gabler Verlag, 2004

[MITSCHELE20] A. Mitschele, „Blockchain”, [Online] Verfügbar unter: https://wirtschaftslexikon.gabler.de/definition/blockchain-54161, Zugriff am: 13. Mai 2020

[BINANCE20] Binance, „Vor- und Nachteile der Blockchain“, [Online] Verfügbar unter: https://www.binance.vision/de/blockchain/positives-and-negatives-of-blockchain, Zugriff am: 13. Mai 2020

[CORRALES19] M. Corrales, M. Fenwick, H. Haapio, „Legal Tech, Smart Contracts and Blockchain”, Singapore, Springer-Verlag, 2019, S. 8

[ZAINUDDIN18] A. Zainuddin, „Token Creation Process“, [Online] Verfügbar unter: https://masterthecrypto.com/wp-content/uploads/2018/10/wsi-imageoptim-token3.png, Zugriff am: 13. Mai 2020

[STREIT20] M. Streit, „Handelsblatt - "Connex-Münze": Fondshaus begibt Immobilien-Token”, [Online] Verfügbar unter: https://www.handelsblatt.com/finanzen/immobilien/blockchain-connex-coin-fondshaus-begibt-immobilien-token/25633406.html?ticket=ST-1661369-teyOY4dgcDbj27DahaXD-ap2, Zugriff am: 13. Mai 2020

[BENDEL18] O. Bendel, „Kryptowährung”, [Online] Verfügbar unter:

https://wirtschaftslexikon.gabler.de/definition/kryptowaehrung-54160/version-277214, Zugriff am: 13. Mai 2020

[LACKES18] R. Lackes, M. Siepermann, I. Sjurts, [Online] Verfügbar unter: https://wirtschaftslexikon.gabler.de/definition/digital-rights-management-drm-29225/version-252838, Zugriff am: 13. Mai 2020

[JUSKALIAN18] R. Juskalian, „Inside the Jordan refugee camp that runs on blockchain", [Online] Verfügbar unter: https://www.technologyreview.com/2018/04/12/143410/inside-the-jordan-refugee-camp-that-runs-on-blockchain/, Zugriff am: 13. Mai 2020

[ETHEREUM20] Ethereum.org, „Was ist Ethereum", [Online] Verfügbar unter: https://ethereum.org/de/what-is-ethereum/, Zugriff am: 13. Mai 2020

[FRANKENFIELD19] J.Frankenfield, „Gas (Ethereum)", [Online] Verfügbar unter: https://www.investopedia.com/terms/g/gas-ethereum.asp, Zugriff am: 13. Mai 2020

[CHITTODA19] J. Chittoda, „Mastering Blockchain Programming with Solidity: Write production-ready smart contracts for Ethereum blockchain with Solidity", Birmingham, Packt Publishing Ltd, 2019, S. 188-210

[APPLEDEVELOPER20] AppleDeveloper, „CryptoKit", [Online] Verfügbar unter: https://developer.apple.com/documentation/cryptokit, Zugriff am: 13. Mai 2020

[KRANKENHAUSIT20] Krankenhaus-IT Journal, Checkliste zur Risikoanalyse in der IT, [Online] Verfügbar unter:

http://www.medizin-edv.de/ARCHIV/CHECKLISTE_ZUR.pdf, Zugriff am: 13. Mai 2020